

Appendix A

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)  
All rights reserved unless otherwise noted

5

File: Ravi Shanker

```
import java.awt.*;
10 import java.applet.*;
import java.lang.*;
import netscape.javascript.*;
import java.io.*;
import java.net.*;
15 import java.util.*;

import com.oreilly.servlet.HttpMessage;

public class ChatClient extends Applet implements ISession
20 {
    netscape.javascript.JSObject      m_JScriptWin;

    public String      m_strMessageHandler;
    public String      m_strErrorHandler;
25 public String      m_EtMh;
    public String      m_SponsorHandler;
    public String      m_LinksHandler;
    public String      m_AdHandler;
    public String      m_CommentsHandler;
30 public String      m_CaptionHandler;
    public String      m_TimeHandler      ;
    public String      m_ClipSource;
    public String      m_isAppletLoad;

35 public Vector CurrentVector ;

    public int StartTime=0;
    public int SponsorCurrent = -1;
    public int LinkCurrent = -1;
40 public int BannerCurrent = -1;
    public int CaptionCurrent = -1;
    public int CommentCurrent = -1;

    public int WhatSec = -1;
45 public void init()
```

```
{
    super.init();

    StartTime=0;
5    SponsorCurrent = -1;
    LinkCurrent = -1;
    BannerCurrent = -1;
    CaptionCurrent = -1;
    CommentCurrent = -1;
10    WhatSec=-1;
        // Get the JavaScript window that will have the various scripts that
        // this applet will call.
        m_JScriptWin = netscape.javascript.JSObject.getWindow(this);
15
        m_strMessageHandler = getParameter("MessageHandler");
        m_EtMh = getParameter("ElapsedTimeMessageHandler");
        m_strErrorHandler = getParameter("ErrorHandler");

        m_SponsorHandler = getParameter("SponsorHandler");
        m_LinksHandler = getParameter("LinksHandler");
        m_AdHandler = getParameter("AdHandler");
        m_CommentsHandler = getParameter("CommentsHandler");
        m_CaptionHandler = getParameter("CaptionHandler");
20        // Set the Javascript to handle if the applet has done it's init()
        m_isAppletLoad = getParameter("isAppletLoad");

        m_TimeHandler = getParameter("TimeHandler");

        m_ClipSource = getParameter("ClipSource");
30
        getNextMessage(0,10,m_ClipSource);
        StartTime=10;

        if (m_JScriptWin != null)
        {
            String Arguments[] = {"Done"};
            m_JScriptWin.call(m_isAppletLoad, Arguments);
            //      System.out.println("setting time here"+seconds);
40        }
    }

    public void setStartTime()
    {
50        int seconds = 0;
    }
}
```

try

```
{
URL url = new URL("http://blanche.tcn.net:8090/servlet/TimeServlet");
HttpMessage msg = new HttpMessage(url);

5      Properties props = new Properties();
      props.put("ClipSource",m_ClipSource);

      InputStream in = msg.sendPostMessage(props);

10     //InputStream in = msg.sendGetMessage();

      DataInputStream data = new DataInputStream(
          new BufferedInputStream(in));
      String lastMessage = data.readLine();
15     seconds = Integer.parseInt(lastMessage);

    }
    catch (SocketException e)
    {
20      // Can't connect to host, report it and wait before trying again
      System.out.println("Can't connect to host: " + e.getMessage());
    }
    catch (FileNotFoundException e)
    {
25      // Servlet doesn't exist, report it and wait before trying again
      System.out.println("Resource not found: " + e.getMessage());
    }
    catch (Exception e)
    {
30      // Some other problem, report it and wait before trying again
      System.out.println("General exception: " +
          e.getClass().getName() + ": " + e.getMessage());
    }
    finally
35    {
        StartTime = seconds ;
        if (m_JScriptWin != null)
        {
            String Arguments[] = {seconds+""};
40            m_JScriptWin.call(m_TimeHandler, Arguments);
            //      System.out.println("setting time here"+seconds);
        }
        getNextMessage(seconds,seconds+10,m_ClipSource);
        StartTime+=10;
45    }
}
```

```
    }

    public void start()
    {
5      }

    public void stop()
    {
10   }

    public int BeginSession(String strAuthor, String strOptions, String strType, String
strMessage)
    {
15      return SendMessage(strAuthor, strType, strMessage);
    }

    public int EndSession(String strAuthor, String strType, String strMessage)
    {
20      return SendMessage(strAuthor, strType, strMessage);
    }

    public int SendMessage(String strAuthor, String strType, String strMessage)
    {
25      if (strAuthor.length() < 0)
          return HandleSessionError(1, "Invalid author specified.");

          if (strMessage.length() < 0)
              return HandleSessionError(2, "Empty message");
30      if (m_strMessageHandler == null)
          {
              m_strMessageHandler = "HandleSessionMessage";
          }
35      if (m_JScriptWin != null)
          {
              String Arguments[] = {strMessage};
              m_JScriptWin.call(m_strMessageHandler, Arguments);
40          }

          return 0;
    }

45 public void SendElapsedTime(String Time)
    {
```



```
int m_time=0;
try
{
    m_time = Integer.parseInt(Time)/1000;
5    }
catch(NumberFormatException nfe)
{
    System.out.println("The Error"+nfe.toString());
    m_time=0;
10    }
//System.out.println("time sent by applet = "+m_time);
HandleEvent(m_time);
//Object Arguments[] = {(new Integer(m_time))};
//    m_JScriptWin.call(m_EtMh, Arguments);
15    //return 0;
}

public void HandleEvent(int ETime)
20    {

    if(WhatSec < ETime)
    {
        Vector _Sponsor = (Vector) CurrentVector.elementAt(0);
        //System.out.println("size of the vector is = "+_Sponsor.size()+"");
        Vector _Links = (Vector) CurrentVector.elementAt(1);
        //System.out.println("size of the links vector is = "+_Links.size()+"");
        Vector _banner = (Vector) CurrentVector.elementAt(2);
30        Vector _caption = (Vector) CurrentVector.elementAt(3);
        Vector _comment = (Vector) CurrentVector.elementAt(4);

        if(_Sponsor != null)
35        {
            for(int i = 0; i < _Sponsor.size();i++)
            {
                Vector Sp1 =(Vector) _Sponsor.elementAt(i);

40                int xy =((Integer)Sp1.elementAt(0)).intValue();
                //System.out.println("xy = "+xy+"");
                if(xy == ETime && SponsorCurrent != xy)
                {
45                    if (m_JScriptWin != null)
                    {
```

```

SponsorCurrent=xy;
String Arguments[] =
{Sp1.elementAt(1)+"" ,Sp1.elementAt(2)+"" ,Sp1.elementAt(3)+""};
m_JScriptWin.call(m_SponsorHandler, Arguments);
5          }
        }
        else if(xy > ETime)
        {
            break;
10        }
    }// end for 1st for
} // end of 1st if

// for links
15 if (_Links != null)
{
    for(int i = 0; i < _Links.size();i++)
    {
20        Vector Sp2 =(Vector) _Links.elementAt(i);

        if(Sp2 != null)
        {
            int xz=((Integer)Sp2.elementAt(0)).intValue();
25 //System.out.println("xy = "+xy+"");
            if(xz == ETime)
            {
                if (m_JScriptWin != null)
                {
30                    //System.out.println("xy = "+xz+"");
                    String msg3="";
                    if( LinkCurrent == xz)
                    {
                        msg3="old";
35                    }
                    else
                    {
                        LinkCurrent=xz;
                        msg3="new";
40                    }
                    String Arguments[] =
{Sp2.elementAt(1)+"" ,Sp2.elementAt(2)+"" ,msg3+""};
                    m_JScriptWin.call(m_LinksHandler,
Arguments);
45                }
            }
        }
    }
}

```

```

                                else if(xz > ETime)
                                {
                                    break;
                                }
5      }
      } // end of 2nd for
    } // end of 1st if
      // for ads
    if (_banner != null)
10   {
        for(int i = 0; i < _banner.size();i++)
        {
            Vector Sp3 =(Vector) _banner.elementAt(i);

15            int az =((Integer)Sp3.elementAt(0)).intValue();
            //System.out.println("xy = "+xy+"");
            if(az == ETime && BannerCurrent != az)
            {
                if (m_JScriptWin != null)
20                {
                    BannerCurrent=az;
                    String Arguments[] =
{Sp3.elementAt(1)+"" ,Sp3.elementAt(2)+"" ,Sp3.elementAt(3)+""};
                    m_JScriptWin.call(m_AdHandler, Arguments);
25                }
            }
            else if(az > ETime)
            {
                break;
30            }
        }
    }

35    // for Caption
    if (_caption != null)
    {
        for(int i = 0; i < _caption.size();i++)
40        {

            Vector Sp4 =(Vector) _caption.elementAt(i);

            if(Sp4 != null)
            {
45                int nn =((Integer)Sp4.elementAt(0)).intValue();
                //System.out.println("xy = "+xy+"");
            }
        }
    }
}
```

```

        if(nn == ETime)
        {
            if (m_JScriptWin != null)
            {
2         //System.out.println("xy = "+xz+"");
                String msg3="";
                if(    CommentCurrent == nn)
                {
                    msg3="old";
10                }
                else
                {
                    CommentCurrent=nn;
                    msg3="new";
15                }
                String Arguments[] = {Sp4.elementAt(1)+""
msg3+""};
                m_JScriptWin.call(m_CaptionHandler,
20 Arguments);
            }
        }
        else if(nn > ETime)
        {
            break;
25        }
    }
} // end of for
} // end of if

30 // for comment
if (_comment != null)
{
    for(int i = 0; i < _comment.size();i++)
35    {

        Vector Sp5 =(Vector) _comment.elementAt(i);

        if(Sp5 != null)
40        {
            int mm =((Integer)Sp5.elementAt(0)).intValue();
            //System.out.println("xy = "+xy+"");
            if(mm == ETime)
            {
45                if (m_JScriptWin != null)
                {

```

```

//System.out.println("xy = "+xz+"");
String msg3="";
if(    CaptionCurrent == mm)
{
    msg3="old";
}
else
{
    CaptionCurrent=mm;
    msg3="new";
}
String Arguments[] = {Sp5.elementAt(1)+"",
msg3+""};
    m_JScriptWin.call(m_CommentsHandler,
15 Arguments);
    }
    }
    else if(mm > ETime)
    {
20         break;
    }
    }
    }// end of for
} // end of if
25
// if the time left for the next call is 1 mins
if(StartTime-ETime <=1)
{
    getNextMessage(StartTime,StartTime+10,m_ClipSource);
30     StartTime=StartTime+10;
}

    WhatSec = ETime;
}
35 }

public void getNextMessage(int _Start, int _End, String ClipSrc)
{
    //System.out.println("in get next message durin init");
40     Object obj=null;
    try
    {
        URL url = new URL("http://blanche.tcn.net:8090/servlet/ILoveTvLiveServlet");

```

```
//URL url = new
URL("http://torlife.wiredkingdom.com/servlet/ILoveTvLiveServlet?ElapsedTimeStart="+_St
art+"&ElapsedTimeEnd="+_End+"&ClipSource="+ClipSrc);
    HttpMessage msg = new HttpMessage(url);
5
        Properties props = new Properties();

        props.put("ElapsedTimeStart",_Start+"");
        props.put("ElapsedTimeEnd",_End+"");
10    props.put("ClipSource",ClipSrc);

        InputStream in = msg.sendPostMessage(props);

        //InputStream in = msg.sendGetMessage();
15
        ObjectInputStream result = new ObjectInputStream(in);

        obj = result.readObject();
    }
20    catch (SocketException e)
    {
        // Can't connect to host, report it and wait before trying again
        System.out.println("Can't connect to host: " + e.getMessage());
    }
25    catch (FileNotFoundException e)
    {
        // Servlet doesn't exist, report it and wait before trying again
        System.out.println("Resource not found: " + e.getMessage());
    }
30    catch (Exception e)
    {
        // Some other problem, report it and wait before trying again
        System.out.println("General exception: " +
            e.getClass().getName() + ": " + e.getMessage());
35    }
    CurrentVector=(Vector)obj;
}

public int HandleSessionError(int iNumber, String strDescription)
40    {
        //System.out.println(iNumber + " : " + strDescription);

        if (m_strErrorHandler == null)
        {
45            m_strErrorHandler = "HandleSessionError";
        }
    }
}
```

```

        if (m_JScriptWin != null)
        {
            String Arguments[] = {Integer.toString(iNumber), strDescription};
5           m_JScriptWin.call(m_strErrorHandler, Arguments);
        }

        return iNumber;
    }
10
}
```

[illegible]

## Appendix B

### Excalibur

#### 5 File

VERSION 5.00

Object = "{648A5603-2C6E-101B-82B6-000000000014}#1.1#0"; "MSCOMM32.OCX"

Object = "{248DD890-BB45-11CF-9ABC-0080C7E7B78D}#1.0#0"; "MSWINSCK.OCX"

10 Begin VB.Form FrmLine21 Watch

    Caption        = "Line21 Watch"

    ClientHeight   = 2655

    ClientLeft     = 60

    ClientTop      = 375

15     ClientWidth   = 11595

    LinkTopic      = "Form1"

    MaxButton      = 0 'False

    MinButton      = 0 'False

    ScaleHeight    = 2655

20     ScaleWidth    = 11595

    StartPosition  = 3 'Windows Default

    Begin VB.CheckBox Field2

        Caption     = "Field 2"

        Enabled     = 0 'False

25     Height       = 255

        Left        = 9360

        TabIndex    = 7

        Top         = 2400

        Width       = 975

30     End

    Begin VB.CheckBox Field1

        Caption     = "Field 1"

        Enabled     = 0 'False

        Height      = 255

35     Left         = 9360

        TabIndex    = 6

        Top         = 2040

        Width       = 855

    End

40     Begin VB.CheckBox Protocol

        Caption     = "Protocol"

        Enabled     = 0 'False

        Height      = 255

        Left        = 8160

45     TabIndex     = 5

        Top         = 2040



```
        Width      = 975
    End
    Begin VB.ComboBox CB_Port
        Height      = 315
    5      Left       = 6960
        TabIndex    = 4
        Text        = "Combo1"
        Top         = 2040
        Width       = 975
    10     End
        Begin VB.TextBox EB_IP
            Height    = 285
            Left      = 5400
            TabIndex  = 3
    15      Top       = 2040
            Width     = 1335
        End
        Begin MSWinsockLib.Winsock Winsock1
            Left      = 960
    20      Top       = 2040
            _ExtentX  = 741
            _ExtentY  = 741
            _Version  = 393216
        End
    25     Begin MSCommLib.MSComm MSComm1
            Left      = 240
            Top       = 2040
            _ExtentX  = 847
            _ExtentY  = 847
    30      _Version  = 393216
            DTREnable = -1 'True
            InputLen  = 1
            RThreshold = 2
            SThreshold = 1
    35     End
        Begin VB.CommandButton B_Watch
            Caption    = "Watch"
            Height     = 495
            Left       = 2520
    40      TabIndex  = 2
            Top        = 2040
            Width      = 1455
        End
        Begin VB.ComboBox CB_ComF1
    45      Height    = 315
            Left      = 4320
```

```

        Style      = 2 'Dropdown List
        TabIndex   = 1
        Top        = 2040
        Width      = 855
5    End
    Begin VB.ListBox LB_Triggers
        BeginProperty Font
            Name      = "Courier New"
            Size      = 8.25
10         Charset   = 0
            Weight    = 400
            Underline  = 0 'False
            Italic     = 0 'False
            Strikethrough = 0 'False
15         EndProperty
            Height    = 1740
            Left      = 120
            TabIndex  = 0
            Top       = 120
20         Width     = 11175
        End
    End
    Attribute VB_Name = "FrmLine21Watch"
    Attribute VB_GlobalNameSpace = False
25    Attribute VB_Creatable = False
    Attribute VB_PredeclaredId = True
    Attribute VB_Exposed = False
    Dim Loading As Boolean
    Dim running As Boolean
30    Dim COMF As String
    Dim SyncFlag As Boolean
    Dim SyncCount As Integer
    Dim mes As String * 1024

35    Option Explicit

    Private Sub B_Watch_Click()
        If running = False Then
            B_Watch.Caption = "Stop"
40         LB_Triggers.Clear
            Line21Decode 0, 0, mes
            SyncFlag = False
            SyncCount = 0
            running = True
45         If COMF <> "LAN" Then
            MSComm1.CommPort = Val(Right(COMF, 1))

```

```

    MSComm1.Settings = "9600,N,8,1"
    MSComm1.PortOpen = True
Else
    Winsock1.RemoteHost = EB_IP.Text
5    Winsock1.RemotePort = CB_Port.Text
    Winsock1.Connect
End If
Else
    B_Watch.Caption = "Watch"
10    running = False
    If COMF <> "LAN" Then
        MSComm1.PortOpen = False
    Else
        Winsock1.Close
15    End If
End If

End Sub

20 Private Sub CB_ComF1_Click()
    Dim I As Integer
    Dim temp As String

    If Loading = False Then
25        Loading = True
        COMF = CB_ComF1.List(CB_ComF1.ListIndex)
        If COMF = "LAN" Then
            EB_IP.Enabled = True
            CB_Port.Enabled = True
30        Else
            EB_IP.Enabled = False
            CB_Port.Enabled = False
        End If
        Loading = False
35    End If

End Sub

Private Sub Form_Load()
40    Dim temp As String
    Dim I As Integer

    Loading = True
    running = False
45    COMF = GetSetting(APPNAME, SETTING, TCOMF, "COM1")
    For I = 1 To 4
```

```
temp = "COM" + Format(I)
CB_ComF1.AddItem (temp)
If COMF = temp Then
    CB_ComF1.ListIndex = CB_ComF1.ListCount - 1
5   End If
Next I
CB_ComF1.AddItem ("LAN")
If COMF = "LAN" Then
    CB_ComF1.ListIndex = CB_ComF1.ListCount - 1
10  EB_IP.Enabled = True
    CB_Port.Enabled = True
Else
    EB_IP.Enabled = False
    CB_Port.Enabled = False
15  End If
EB_IP.Text = GetSetting(APPNAME, SETTING, IP, "223.223.223.120")
For I = 2000 To 9000 Step 1000
    CB_Port.AddItem (Format(I))
Next I
20  CB_Port.Text = GetSetting(APPNAME, SETTING, PORT, "3000")
    Loading = False

End Sub

25  Private Sub Form_Unload(Cancel As Integer)
    SaveSetting APPNAME, SETTING, TCOMF, COMF
    SaveSetting APPNAME, SETTING, IP, EB_IP.Text
    SaveSetting APPNAME, SETTING, PORT, CB_Port.Text

30  End Sub

Private Sub MSComm1_OnComm()
    Dim Trig As Long
    Dim status As Integer
    Static Num As Integer
    Static Dat As String
    Dim C0 As Long
    Dim C1 As Long
    Dim C2 As Long
40
    status = MSComm1.CommEvent
    If status = 2 And running = True Then
        If SyncFlag = False Then      'Getting is sync is real ugly
            SyncCount = SyncCount + 1  'We watch for the protocol characters
45  If SyncCount = 1 Then
            MSComm1.InBufferCount = 0
```

```
Else
    Dat = MSComm1.Input          'and keep track of whether data is from
    Select Case Asc(Dat)         'field 1, field 2 or both.
        Case 17, 129            'We want to start capturing data on field 1
5         Protocol.Value = 1    'if both fields or at the start of the
            If Field1.Value = 0 And Field2.Value = 1 Then
                Num = 3
            Else
                Num = 0
10         End If
            Field1.Value = 1      'field data.
        Case 18, 130
            Protocol.Value = 1
            Num = 0
15         Field2.Value = 1
        Case 148, 28            'If it is not protocol data then we must
            If SyncCount > 2 And Protocol.Value = 0 Then
                Field1.Value = 1  'decide on the field based on the control
                Num = 1          'codes.
20         End If
        Case 21, 157, 1, 2, 131, 4, 133, 134, 7, 8, 137, 138, 11, 140, 13, 14, 143
            If SyncCount > 2 And Protocol.Value = 0 Then
                Field2.Value = 1
                Num = 1
25         End If
        Case Else
            Num = Num + 1
        End Select
    If SyncCount > 5 And (Field1.Value = 1 Or Field2.Value = 1) And Num = 2 Then
30         SyncFlag = True
            If Protocol.Value = 1 Then
                MSComm1.InputLen = 3
            Else
                MSComm1.InputLen = 2
35         End If
    End If
End If
Else
    Dat = MSComm1.Input
40     If Protocol.Value = 1 Then
        C0 = CByte(Asc(Left(Dat, 1)))
        C1 = CByte(Asc(Mid(Dat, 2, 1))) And 127
        C2 = CByte(Asc(Right(Dat, 1))) And 127
    Else
45     If Field1.Value = 1 Then
        C0 = 129
```

```
Else
    C0 = 130
End If
End If
5  Select Case C0
    Case 129, 17
        Trig = Line21Decode(1, C1, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
10        End If
        Trig = Line21Decode(1, C2, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
15    Case 130, 18
        Trig = Line21Decode(2, C1, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
20    Trig = Line21Decode(2, C2, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
    End Select
25    End If
    End If
End Sub

30 Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long)
    Static InputLen As Integer
    Static Num As Integer
    Static Dat As String
    Dim Trig As Long
35    Dim C0 As Long
    Dim C1 As Long
    Dim C2 As Long

    If running = True Then
40        If InputLen < 1 Or InputLen > 3 Then
            InputLen = 1
        End If
        While Winsock1.BytesReceived >= InputLen
            If SyncFlag = False Then        'Getting in sync is real ugly
45                SyncCount = SyncCount + 1    'We watch for the protocol characters
            If SyncCount = 1 Then
```

```
MSComm1.InBufferCount = 0
Else
Winsock1.GetData Dat, , 1      'and keep track of whether data is from
Select Case Asc(Dat)          'field 1, field 2 or both.
5   Case 17, 129              'We want to start capturing data on field 1
    Protocol.Value = 1        'if both fields or at the start of the
    If Field1.Value = 0 And Field2.Value = 1 Then
        Num = 3
    Else
10    Num = 0
    End If
    Field1.Value = 1          'field data.
    Case 18, 130
        Protocol.Value = 1
15    Num = 0
        Field2.Value = 1
    Case 148, 28              'If it is not protocol data then we must
    If SyncCount > 2 And Protocol.Value = 0 Then
        Field1.Value = 1      'decide on the field based on the control
20    Num = 1                  'codes.
    End If
    Case 21, 157, 1, 2, 131, 4, 133, 134, 7, 8, 137, 138, 11, 140, 13, 14, 143
    If SyncCount > 2 And Protocol.Value = 0 Then
        Field2.Value = 1
25    Num = 1
    End If
    Case Else
        Num = Num + 1
    End Select
30 If SyncCount > 5 And (Field1.Value = 1 Or Field2.Value = 1) And Num = 2 Then
    SyncFlag = True
    If Protocol.Value = 1 Then
        InputLen = 3
    Else
35    InputLen = 2
    End If
    End If
    End If
Else
40 Winsock1.GetData Dat, , InputLen
    If Protocol.Value = 1 Then
        C0 = CByte(Asc(Left(Dat, 1)))
        C1 = CByte(Asc(Mid(Dat, 2, 1))) And 127
        C2 = CByte(Asc(Right(Dat, 1))) And 127
45 Else
    If Field1.Value = 1 Then
```

```

    C0 = 129
    Else
    C0 = 130
    End If
5  End If
    Select Case C0
    Case 129, 17
        Trig = Line21Decode(1, C1, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
10         LB_Triggers.AddItem (mes)
        End If
        Trig = Line21Decode(1, C2, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
15         LB_Triggers.AddItem (mes)
        End If
    Case 130, 18
        Trig = Line21Decode(2, C1, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
20         LB_Triggers.AddItem (mes)
        End If
        Trig = Line21Decode(2, C2, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
25         LB_Triggers.AddItem (mes)
        End If
    End Select
    End If
    Wend
    End If
30 End Sub
```



Appendix B

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)  
All rights reserved

5

Excalibur

File: FrmLine21Watch.frm

```
10  VERSION 5.00
    Object = "{648A5603-2C6E-101B-82B6-000000000014}#1.1#0"; "MSCOMM32.OCX"
    Object = "{248DD890-BB45-11CF-9ABC-0080C7E7B78D}#1.0#0"; "MSWINSCK.OCX"
    Begin VB.Form FrmLine21Watch
        Caption      = "Line21 Watch"
15      ClientHeight = 2655
        ClientLeft   = 60
        ClientTop    = 375
        ClientWidth  = 11595
        LinkTopic    = "Form1"
20      MaxButton    = 0 'False
        MinButton    = 0 'False
        ScaleHeight   = 2655
        ScaleWidth    = 11595
        StartUpPosition = 3 'Windows Default
25      Begin VB.CheckBox Field2
        Caption      = "Field 2"
        Enabled     = 0 'False
        Height       = 255
        Left         = 9360
30      TabIndex     = 7
        Top          = 2400
        Width        = 975
    End
    Begin VB.CheckBox Field1
35      Caption      = "Field 1"
        Enabled     = 0 'False
        Height       = 255
        Left         = 9360
        TabIndex     = 6
40      Top          = 2040
        Width        = 855
    End
    Begin VB.CheckBox Protocol
45      Caption      = "Protocol"
        Enabled     = 0 'False
        Height       = 255
```

```

    Left      = 8160
    TabIndex  = 5
    Top       = 2040
    Width     = 975
5  End
Begin VB.ComboBox CB_Port
    Height     = 315
    Left       = 6960
    TabIndex   = 4
10   Text     = "Combo1"
    Top        = 2040
    Width      = 975
End
Begin VB.TextBox EB_IP
15   Height   = 285
    Left      = 5400
    TabIndex  = 3
    Top       = 2040
    Width     = 1335
20 End
Begin MSWinsockLib.Winsock Winsock1
    Left      = 960
    Top       = 2040
    _ExtentX  = 741
25   _ExtentY = 741
    _Version  = 393216
End
Begin MSCommLib.MSComm MSComm1
30   Left     = 240
    Top       = 2040
    _ExtentX  = 847
    _ExtentY  = 847
    _Version  = 393216
    DTREnable = -1 'True
35   InputLen = 1
    RThreshold = 2
    SThreshold = 1
End
Begin VB.CommandButton B_Watch
40   Caption  = "Watch"
    Height    = 495
    Left     = 2520
    TabIndex = 2
    Top      = 2040
45   Width   = 1455
End
```

```
Begin VB.ComboBox CB_ComF1
    Height    = 315
    Left      = 4320
    Style     = 2 'Dropdown List
5    TabIndex = 1
    Top      = 2040
    Width    = 855
End
Begin VB.ListBox LB_Triggers
10    BeginProperty Font
        Name      = "Courier New"
        Size     = 8.25
        Charset  = 0
        Weight   = 400
15    Underline = 0 'False
        Italic    = 0 'False
        Strikethrough = 0 'False
    EndProperty
    Height    = 1740
20    Left     = 120
    TabIndex  = 0
    Top      = 120
    Width    = 11175
    End
25    End
    Attribute VB_Name = "FrmLine21Watch"
    Attribute VB_GlobalNameSpace = False
    Attribute VB_Creatable = False
    Attribute VB_PredeclaredId = True
30    Attribute VB_Exposed = False
    Dim Loading As Boolean
    Dim running As Boolean
    Dim COMF As String
    Dim SyncFlag As Boolean
35    Dim SyncCount As Integer
    Dim mes As String * 1024

Option Explicit

40    Private Sub B_Watch_Click()
        If running = False Then
            B_Watch.Caption = "Stop"
            LB_Triggers.Clear
            Line21Decode 0, 0, mes
45    SyncFlag = False
            SyncCount = 0
```

```
running = True
If COMF <> "LAN" Then
    MSComm1.CommPort = Val(Right(COMF, 1))
    MSComm1.Settings = "9600,N,8,1"
5    MSComm1.PortOpen = True
Else
    Winsock1.RemoteHost = EB_IP.Text
    Winsock1.RemotePort = CB_Port.Text
    Winsock1.Connect
10    End If
Else
    B_Watch.Caption = "Watch"
    running = False
    If COMF <> "LAN" Then
15        MSComm1.PortOpen = False
    Else
        Winsock1.Close
    End If
End If
20    End Sub

Private Sub CB_ComF1_Click()
    Dim I As Integer
25    Dim temp As String

    If Loading = False Then
        Loading = True
        COMF = CB_ComF1.List(CB_ComF1.ListIndex)
30    If COMF = "LAN" Then
        EB_IP.Enabled = True
        CB_Port.Enabled = True
    Else
        EB_IP.Enabled = False
35        CB_Port.Enabled = False
    End If
    Loading = False
End If

40    End Sub

Private Sub Form_Load()
    Dim temp As String
    Dim I As Integer
45    Loading = True
```

```
running = False
COMF = GetSetting(APPNAME, SETTING, TCOMF, "COM1")
For I = 1 To 4
    temp = "COM" + Format(I)
5    CB_ComF1.AddItem (temp)
    If COMF = temp Then
        CB_ComF1.ListIndex = CB_ComF1.ListCount - 1
    End If
Next I
10    CB_ComF1.AddItem ("LAN")
    If COMF = "LAN" Then
        CB_ComF1.ListIndex = CB_ComF1.ListCount - 1
        EB_IP.Enabled = True
        CB_Port.Enabled = True
15    Else
        EB_IP.Enabled = False
        CB_Port.Enabled = False
    End If
    EB_IP.Text = GetSetting(APPNAME, SETTING, IP, "223.223.223.120")
20    For I = 2000 To 9000 Step 1000
        CB_Port.AddItem (Format(I))
    Next I
    CB_Port.Text = GetSetting(APPNAME, SETTING, PORT, "3000")
    Loading = False
25    End Sub

Private Sub Form_Unload(Cancel As Integer)
    SaveSetting APPNAME, SETTING, TCOMF, COMF
30    SaveSetting APPNAME, SETTING, IP, EB_IP.Text
    SaveSetting APPNAME, SETTING, PORT, CB_Port.Text

End Sub

35    Private Sub MSComm1_OnComm()
        Dim Trig As Long
        Dim status As Integer
        Static Num As Integer
        Static Dat As String
40        Dim C0 As Long
        Dim C1 As Long
        Dim C2 As Long

        status = MSComm1.CommEvent
45        If status = 2 And running = True Then
            If SyncFlag = False Then 'Getting is sync is real ugly
```

```
SyncCount = SyncCount + 1      'We watch for the protocol characters
If SyncCount = 1 Then
    MSComm1.InBufferCount = 0
Else
5   Dat = MSComm1.Input        'and keep track of whether data is from
    Select Case Asc(Dat)      'field 1, field 2 or both.
        Case 17, 129         'We want to start capturing data on field 1
            Protocol.Value = 1 'if both fields or at the start of the
            If Field1.Value = 0 And Field2.Value = 1 Then
10              Num = 3
            Else
                Num = 0
            End If
            Field1.Value = 1   'field data.
15         Case 18, 130
            Protocol.Value = 1
            Num = 0
            Field2.Value = 1
            Case 148, 28      'If it is not protocol data then we must
20             If SyncCount > 2 And Protocol.Value = 0 Then
                Field1.Value = 1 'decide on the field based on the control
                Num = 1         'codes.
            End If
            Case 21, 157, 1, 2, 131, 4, 133, 134, 7, 8, 137, 138, 11, 140, 13, 14, 143
25             If SyncCount > 2 And Protocol.Value = 0 Then
                Field2.Value = 1
                Num = 1
            End If
            Case Else
30             Num = Num + 1
        End Select
        If SyncCount > 5 And (Field1.Value = 1 Or Field2.Value = 1) And Num = 2 Then
            SyncFlag = True
            If Protocol.Value = 1 Then
35             MSComm1.InputLen = 3
            Else
                MSComm1.InputLen = 2
            End If
        End If
40    End If
    Else
        Dat = MSComm1.Input
        If Protocol.Value = 1 Then
            C0 = CByte(Asc(Left(Dat, 1)))
            C1 = CByte(Asc(Mid(Dat, 2, 1))) And 127
45            C2 = CByte(Asc(Right(Dat, 1))) And 127
```

```
Else
  If Field1.Value = 1 Then
    C0 = 129
  Else
5    C0 = 130
  End If
End If
Select Case C0
  Case 129, 17
10    Trig = Line21Decode(1, C1, mes)
    If Mid(mes, 1, 1) <> Chr(0) Then
      LB_Triggers.AddItem (mes)
    End If
    Trig = Line21Decode(1, C2, mes)
15    If Mid(mes, 1, 1) <> Chr(0) Then
      LB_Triggers.AddItem (mes)
    End If
  Case 130, 18
    Trig = Line21Decode(2, C1, mes)
20    If Mid(mes, 1, 1) <> Chr(0) Then
      LB_Triggers.AddItem (mes)
    End If
    Trig = Line21Decode(2, C2, mes)
    If Mid(mes, 1, 1) <> Chr(0) Then
25    LB_Triggers.AddItem (mes)
    End If
  End Select
End If
End If
30 End Sub
```

```
Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long)
  Static InputLen As Integer
35  Static Num As Integer
  Static Dat As String
  Dim Trig As Long
  Dim C0 As Long
  Dim C1 As Long
40  Dim C2 As Long

  If running = True Then
    If InputLen < 1 Or InputLen > 3 Then
      InputLen = 1
45  End If
    While Winsock1.BytesReceived >= InputLen
```

```
If SyncFlag = False Then      'Getting in sync is real ugly
    SyncCount = SyncCount + 1  'We watch for the protocol characters
    If SyncCount = 1 Then
        MSComm1.InBufferCount = 0
5    Else
        Winsock1.GetData Dat, , 1      'and keep track of whether data is from
        Select Case Asc(Dat)           'field 1, field 2 or both.
            Case 17, 129                'We want to start capturing data on field 1
                Protocol.Value = 1      'if both fields or at the start of the
10            If Field1.Value = 0 And Field2.Value = 1 Then
                Num = 3
            Else
                Num = 0
            End If
15            Field1.Value = 1          'field data.
            Case 18, 130
                Protocol.Value = 1
                Num = 0
                Field2.Value = 1
20            Case 148, 28              'If it is not protocol data then we must
                If SyncCount > 2 And Protocol.Value = 0 Then
                    Field1.Value = 1    'decide on the field based on the control
                    Num = 1             'codes.
                End If
25            Case 21, 157, 1, 2, 131, 4, 133, 134, 7, 8, 137, 138, 11, 140, 13, 14, 143
                If SyncCount > 2 And Protocol.Value = 0 Then
                    Field2.Value = 1
                    Num = 1
                End If
30            Case Else
                Num = Num + 1
            End Select
            If SyncCount > 5 And (Field1.Value = 1 Or Field2.Value = 1) And Num = 2 Then
                SyncFlag = True
35            If Protocol.Value = 1 Then
                InputLen = 3
            Else
                InputLen = 2
            End If
40            End If
        End If
    Else
        Winsock1.GetData Dat, , InputLen
        If Protocol.Value = 1 Then
45            C0 = CByte(Asc(Left(Dat, 1)))
            C1 = CByte(Asc(Mid(Dat, 2, 1))) And 127
```



```

    C2 = CByte(Asc(Right(Dat, 1))) And 127
Else
    If Field1.Value = 1 Then
        C0 = 129
5      Else
        C0 = 130
    End If
End If
Select Case C0
10   Case 129, 17
        Trig = Line21Decode(1, C1, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
15     Trig = Line21Decode(1, C2, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
        Case 130, 18
20     Trig = Line21Decode(2, C1, mes)
        If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
        Trig = Line21Decode(2, C2, mes)
25     If Mid(mes, 1, 1) <> Chr(0) Then
            LB_Triggers.AddItem (mes)
        End If
    End Select
End If
30   Wend
End If

End Sub

35
```

File: FrmLine21Watch.log

- 5 Line 15: Class MSCommLib.MSComm of control MSComm1 was not a loaded control class.  
Line 18: The property name \_ExtentX in MSComm1 is invalid.  
Line 19: The property name \_ExtentY in MSComm1 is invalid.  
Line 20: The property name \_Version in MSComm1 is invalid.  
Line 21: The property name DTREnable in MSComm1 is invalid.
- 10 Line 22: The property name RThreshold in MSComm1 is invalid.

Microsoft Visual Studio .NET 2003  
Project: FrmLine21Watch.log  
File: FrmLine21Watch.log  
Line: 15  
Column: 1  
Error: Class MSCommLib.MSComm of control MSComm1 was not a loaded control class.  
Line: 18  
Column: 1  
Error: The property name \_ExtentX in MSComm1 is invalid.  
Line: 19  
Column: 1  
Error: The property name \_ExtentY in MSComm1 is invalid.  
Line: 20  
Column: 1  
Error: The property name \_Version in MSComm1 is invalid.  
Line: 21  
Column: 1  
Error: The property name DTREnable in MSComm1 is invalid.  
Line: 22  
Column: 1  
Error: The property name RThreshold in MSComm1 is invalid.

File: Funt.c

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)

5 All rights reserved

```

#include <windows.h>
10 #include <stdio.h>
#include <string.h>
#include "L21Dec.h"
#include "sMachine.h"

15 extern char MaxLength[CLASSMAX][RECORDMAX];
extern char MinLength[CLASSMAX][RECORDMAX];
extern char XDSMes[CLASSMAX][RECORDMAX][MESLEN];
extern void (*XDSDisp[CLASSMAX][RECORDMAX])();
extern char *CType[];

20 // When we start an XDS record we must remember which class it is from

int XDSBegin(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
25   cStatePtr->XDSClass = (ch - 1) / 2;
   return(TRUE);
}

// We art starting the XDS record so we must remember the record. If the record is
30 // already open then we should give an overwrite error. Prepare the XDS data structure
// for the record.

int XDSSStart(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
35   cStatePtr->XDSRecord = ch;
   if(cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Open == TRUE)
   {
       strcpy(mes,XDSSSTART);
       strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
40   strcat(mes,OVERWRITE);
   }
   cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Open = TRUE;
   cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length = 0;
   cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Checksum = cStatePtr-
45 >XDSClass * 2 + 1 + ch;
   return(TRUE);
}
```

```
    }

    // When we continue an XDS record we must save the record type. We check to see if the
    // record is open. If it is not then we give an error message and open it as if this
5    // was a start.

    int XDSCont(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
    {
        cStatePtr->XDSRecord = ch;
10    if(cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Open == FALSE)
        {
            strcpy(mes,XDSSTART);
            strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
            strcat(mes,CONTNOSTART);
15    cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Open = TRUE;
            cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length = 0;
            cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Checksum = cStatePtr-
            >XDSClass * 2 + 1 + ch;
        }
20    return(TRUE);
    }

    // When a character arrives for the XDS record we must add it to the checksum. We check
    // to see if this makes the record longer than the max length for this record. If not
25    // then we store the data for this record and increment the record length.

    int XDSData(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
    {
        cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Checksum += ch;
30    if(cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length <
        MaxLength[cStatePtr->XDSClass][cStatePtr->XDSRecord])
        cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Data[cStatePtr-
        >XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length] = ch;
        cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length++;
35    return(TRUE);
    }

    // When we close an XDS record we must check to see that we have received enough data,
    // that we have not received too much data and that the checksum is valid. Any of these
40    // will cause error messages. If the record is OK then display the XDS data. This is
    // done through a look-up table of routines so each XDS record can be displayed
    // differently.

    int XDSEnd(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
45    {
        cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Open = FALSE;
```

```
if(cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length <
MinLength[cStatePtr->XDSClass][cStatePtr->XDSRecord])
{
    strcpy(mes,XDSSTART);
5    strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
    strcat(mes,TOOSHORT);
}
else if(cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length >
MaxLength[cStatePtr->XDSClass][cStatePtr->XDSRecord])
10 {
    strcpy(mes,XDSSTART);
    strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
    strcat(mes,TOOLONG);
}
15 else
{
    if(((cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Checksum +
XSEND + ch) & 0x7f) != 0)
    {
20        strcpy(mes,XDSSTART);
        strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
        strcat(mes,BADCHECKSUM);
    }
    else
25        XDSDisp[cStatePtr->XDSClass][cStatePtr->XDSRecord](mes, cStatePtr);
}
return(TRUE);
}

30 // When we get a control character (0x10 to 0x1f) we must save it so we can use
// it as part of a control pair with the next character.

int ControlSave(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
35    cStatePtr->CurrentCtrl[field - 1] = ch;
    return(TRUE);
}

// For each of the four possible streams for each field we must maintain a buffer of
40 // the characters that have arrived for the stream. We will detect the "end of a caption
// and dump the caption as a message.

int DataSave(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
45    int f, c, m;
```

```
f = field - 1;
c = cStatePtr->Channel[f];
m = cStatePtr->Mode[f];
if(cStatePtr->TextLength[f][c][m] < 255)
5   {
    cStatePtr->Text[f][c][m][cStatePtr->TextLength[f][c][m]] = ch;
    cStatePtr->TextLength[f][c][m]++;
  }
return(TRUE);
10 }

// There are different types of captions. We store the type of caption for the stream.

int PopOnDecode(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
15 {
    int f;

    f = field - 1;
    cStatePtr->Mode[f] = CAP;
20   if(cStatePtr->CurrentCtrl[f] < 0x18)
    {
        cStatePtr->Channel[f] = C1;
        cStatePtr->Type[f][C1] = POPUP;
    }
25   else
    {
        cStatePtr->Channel[f] = C2;
        cStatePtr->Type[f][C2] = POPUP;
    }
30   return(TRUE);
}

int RollUpDecode(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
35 {
    int f, c;

    f = field - 1;
    cStatePtr->Mode[f] = CAP;
    if(cStatePtr->CurrentCtrl[f] < 0x18)
40     c = C1;
    else
        c = C2;
    cStatePtr->Channel[f] = c;
    if(ch == 0x25)
45     cStatePtr->Type[f][c] = ROLLUP2;
    if(ch == 0x26)
```

```

    cStatePtr->Type[f][c] = ROLLUP3;
    if(ch == 0x27)
        cStatePtr->Type[f][c] = ROLLUP4;
    return(TRUE);
5   }

int PaintOnDecode(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    int f;
10   f = field - 1;
    cStatePtr->Mode[f] = CAP;
    if(cStatePtr->CurrentCtrl[f] < 0x18)
    {
15     cStatePtr->Channel[f] = C1;
        cStatePtr->Type[f][C1] = PAINTON;
    }
    else
    {
20     cStatePtr->Channel[f] = C2;
        cStatePtr->Type[f][C2] = PAINTON;
    }
    return(TRUE);
}

25 int EndOfCaptionDecode(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    int f;

30   f = field - 1;
    cStatePtr->Mode[f] = CAP;
    if(cStatePtr->CurrentCtrl[f] < 0x18)
    {
35     cStatePtr->Channel[f] = C1;
        cStatePtr->Type[f][C1] = POPUP;
    }
    else
    {
40     cStatePtr->Channel[f] = C2;
        cStatePtr->Type[f][C2] = POPUP;
    }
    return(TRUE);
}

45 // If the stream is not a caption stream then it is a text stream and should be marked
    // as such. The type does not apply to the stream if it is a text stream.
```

```
int TextDecode(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    int f;
5    f = field - 1;
    cStatePtr->Mode[f] = TXT;
    if(cStatePtr->CurrentCtrl[f] < 0x18)
        cStatePtr->Channel[f] = C1;
10    else
        cStatePtr->Channel[f] = C2;
    return(TRUE);
}

15 // URLs are a special use of the T2 stream. We track the T2 stream outside of the
    // regular text streams so that we can monitor it for the URL checksum which we use
    // to determine the end of the URL.

int URLDecode(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
20 {
    cStatePtr->Mode[0] = TXT;
    cStatePtr->Channel[0] = C2;
    return(TRUE);
}

25 // This is a tricky one. It is possible to change streams with a mid row code. In this
    // case we change where we put text.

int ChannelChange(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
30 {
    int f;

    f = field - 1;
    if((ch >= 0x20) && (ch <= 0x3f))
35 {
        if(cStatePtr->CurrentCtrl[f] < 0x18)
            cStatePtr->Channel[f] = C1;
        else
            cStatePtr->Channel[f] = C2;
40 }
    return(TRUE);
}

45 // This is where we figure out what the current stream is and create a message which
    // includes the stream type, the caption type (if applicable) and the text we have
    // captured for this stream. We will call this if: we get a control pair carriage
```



```
// return, we get a positioning code pair (PAC) or we see a checksum in the URL
// stream.
```

```
int DumpText(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
5   {
    int f, c, m;

    f = field - 1;
    m = cStatePtr->Mode[f];
10   c = cStatePtr->Channel[f];
    if(cStatePtr->TextLength[f][c][m] > 0)
    {
        if(field == FIELD1)
        {
15         if(m == CAP)
            {
                if(c == C1)
                {
                    strcpy(mes, C1START);
20                 strcat(mes, CType[cStatePtr->Type[f][c]]);
                }
                else
                {
                    strcpy(mes, C2START);
25                 strcat(mes, CType[cStatePtr->Type[f][c]]);
                }
            }
        else
        {
30         if(c == C1)
            {
                strcpy(mes, T1START);
                strcat(mes, SEVENSP);
            }
35         else
            {
                strcpy(mes, T2START);
                strcat(mes, SEVENSP);
            }
40         }
        }
    else
    {
        if(m == CAP)
45         {
            if(c == C1)
```

```

    {
        strcpy(mes, C3START);
        strcat(mes, CType[cStatePtr->Type[f][c]]);
    }
5   else
    {
        strcpy(mes, C4START);
        strcat(mes, CType[cStatePtr->Type[f][c]]);
    }
10  }
    else
    {
        if(c == C1)
        {
15      strcpy(mes, T3START);
          strcat(mes, SEVENSP);
        }
        else
        {
20      strcpy(mes, T4START);
          strcat(mes, SEVENSP);
        }
    }
25  cStatePtr->Text[f][c][m][cStatePtr->TextLength[f][c][m]] = 0;
    strcat(mes, cStatePtr->Text[f][c][m]);
    cStatePtr->TextLength[f][c][m] = 0;
    }
    return(TRUE);
30  }
```

File: L21Dec.c

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)

5 All rights reserved

```
#include "windows.h"
#include "stdio.h"
10 #include "L21Dec.h"
#include "sMachine.h"

extern SVAL DataState[];
extern unsigned char parity[];

15 struct cState CCstate;
struct cState *cStatePtr;

#pragma warning(disable : 4133)

20 BOOL APIENTRY DllMain(HANDLE hInst, DWORD ul_reason_being_called, LPVOID
lpReserved)
{
    return 1;
25 UNREFERENCED_PARAMETER(hInst);
UNREFERENCED_PARAMETER(ul_reason_being_called);
UNREFERENCED_PARAMETER(lpReserved);
}

30 __declspec(dllexport) LONG APIENTRY Line21Decode(long field, unsigned char ch, char
mes[])
{
    int i, j;

35 if(field == 0) // Reset the finite state machine
    {
        cStatePtr = &CCstate;
        cStatePtr->NextState[0] = DataState;
        cStatePtr->NextState[1] = DataState;
40 cStatePtr->Channel[0] = C1;
        cStatePtr->Channel[1] = C1;
        cStatePtr->Mode[0] = CAP;
        cStatePtr->Mode[1] = CAP;
        for(i = 0; i < CLASSMAX; i++)
45     {
        for(j = 0; j < RECORDMAX; j++)
```

```
        cStatePtr->XDS[i][j].Open = FALSE;
    }
}
else if((field == 2) || (field == 1))
5   {
    ch &= 0x7f;
    mes[0] = 0;
    stateMachine(field, ch, mes);
    }
10  return(0);
    }

void stateMachine(long field, unsigned char ch, char mes[])
15  {
    SVAL *cstate;

    /*
    * This is the state machine driver. Each possible branch of a state
    * starts with a test. If the test is passed then the functions of that
    * branch are executed. The end of functions is indicated by a NULL
    * pointer. After the NULL pointer is a pointer to the next state. The
    * whole state is a single array containing sets of, the state test,
    * the functions for that state branch and the next state. The test of the last
    * set must be the default test which always returns true.
    */
    cstate = cStatePtr->NextState[field - 1];
    while(TRUE)
    {
    30  if(cstate++->fptr(field, ch, mes, cStatePtr)) // Do the test for this branch
        {
            while(*cstate->fptr) // Passed so execute all its functions
            {
                cstate++->fptr(field, ch, mes, cStatePtr);
            }
    35  cStatePtr->NextState[field - 1] = (cstate + 1)->dptr; // Then return pointer to the next state
            break;
        }
        else // Failed this branch test so
    40  {
            while(*cstate++->fptr) // Skip over its functions
            ;
            cstate++; // Skip the state change to the
            // start of the next branch set
    45  }
    }
}
```

```
__declspec(dllexport) LONG APIENTRY ShiftRight(unsigned long value, long shift)
{
5   return(value >> shift);
}

__declspec(dllexport) LONG APIENTRY ShiftLeft(unsigned long value, long shift)
{
10  return(value << shift);
}
```

Microsoft Visual C++ 6.0

File: L21Dec.def

LIBRARY Line21

5

CODE PRELOAD MOVEABLE DISCARDABLE  
DATA PRELOAD SINGLE

10 EXPORTS  
    Line21Decode  
    ShiftRight  
    ShiftLeft

15

Line21Decode  
ShiftRight  
ShiftLeft

File: L21Dec.h

```
__declspec(dllexport) LONG APIENTRY Line21Decode(long field, unsigned char ch, char
mess[]);
5  __declspec(dllexport) LONG APIENTRY ShiftRight(unsigned long, long);
   __declspec(dllexport) LONG APIENTRY ShiftLeft(unsigned long, long);

void stateMachine(long field, unsigned char ch, char mes[]);
void Def(char mes[], struct cState *);
10 void Vcp(char mes[], struct cState *);
   void Typ(char mes[], struct cState *);

int XDSSBegin(long field, unsigned char ch, char mes[], struct cState *);
int XDSSStart(long field, unsigned char ch, char mes[], struct cState *);
15 int XDSCont(long field, unsigned char ch, char mes[], struct cState *);
   int XDSSStart(long field, unsigned char ch, char mes[], struct cState *);
   int XDSDData(long field, unsigned char ch, char mes[], struct cState *);
   int XDSEnd(long field, unsigned char ch, char mes[], struct cState *);
   int ChannelChange(long field, unsigned char ch, char mes[], struct cState *);
20 int DumpText(long field, unsigned char ch, char mes[], struct cState *);
   int ControlSave(long field, unsigned char ch, char mes[], struct cState *);
   int PopOnDecode(long field, unsigned char ch, char mes[], struct cState *);
   int DataSave(long field, unsigned char ch, char mes[], struct cState *);
   int RollUpDecode(long field, unsigned char ch, char mes[], struct cState *);
25 int PaintOnDecode(long field, unsigned char ch, char mes[], struct cState *);
   int TextDecode(long field, unsigned char ch, char mes[], struct cState *);
   int URLDecode(long field, unsigned char ch, char mes[], struct cState *);

int DefaultTest(long field, unsigned char ch, char mes[], struct cState *);
30 int NullTest(long field, unsigned char ch, char mes[], struct cState *);
   int FIXDSTest(long field, unsigned char ch, char mes[], struct cState *);
   int ControlTest(long field, unsigned char ch, char mes[], struct cState *);
   int XDSSStartTest(long field, unsigned char ch, char mes[], struct cState *);
   int XDSContTest(long field, unsigned char ch, char mes[], struct cState *);
35 int XDSEndTest(long field, unsigned char ch, char mes[], struct cState *);
   int RollUpTest(long field, unsigned char ch, char mes[], struct cState *);
   int PopOnTest(long field, unsigned char ch, char mes[], struct cState *);
   int PaintOnTest(long field, unsigned char ch, char mes[], struct cState *);
   int EndOfCaptionTest(long field, unsigned char ch, char mes[], struct cState *);
40 int TextModeTest(long field, unsigned char ch, char mes[], struct cState *);
   int PACTest(long field, unsigned char ch, char mes[], struct cState *);
   int CRTTest(long field, unsigned char ch, char mes[], struct cState *);
   int CtrlChangeTest(long field, unsigned char ch, char mes[], struct cState *);
   int URLTest(long field, unsigned char ch, char mes[], struct cState *);
45 int URLEndTest(long field, unsigned char ch, char mes[], struct cState *);
```

File: Line21.mak

```
# Microsoft Developer Studio Generated NMAKE File, Format Version 40001
# ** DO NOT EDIT **

5  # TARGETTYPE "Win32 (x86) Dynamic-Link Library" 0x0102

    !IF "$(CFG)" == ""
        CFG=Line21 - Win32 Debug
10  !MESSAGE No configuration specified. Defaulting to Line21 - Win32 Debug.
    !ENDIF

    !IF "$(CFG)" != "Line21 - Win32 Release" && "$(CFG)" != "Line21 - Win32 Debug"
        !MESSAGE Invalid configuration "$(CFG)" specified.
15  !MESSAGE You can specify a configuration when running NMAKE on this makefile
        !MESSAGE by defining the macro CFG on the command line. For example:
        !MESSAGE
        !MESSAGE NMAKE /f "Line21.mak" CFG="Line21 - Win32 Debug"
        !MESSAGE
20  !MESSAGE Possible choices for configuration are:
        !MESSAGE
        !MESSAGE "Line21 - Win32 Release" (based on "Win32 (x86) Dynamic-Link Library")
        !MESSAGE "Line21 - Win32 Debug" (based on "Win32 (x86) Dynamic-Link Library")
        !MESSAGE
25  !ERROR An invalid configuration is specified.
    !ENDIF

    !IF "$(OS)" == "Windows_NT"
        NULL=
30  !ELSE
        NULL=nul
    !ENDIF

#####
#####
35  # Begin Project
    # PROP Target_Last_Scanned "Line21 - Win32 Debug"
    RSC=rc.exe
    MTL=mktyplib.exe
    CPP=cl.exe
40

    !IF "$(CFG)" == "Line21 - Win32 Release"

        # PROP BASE Use_MFC 0
        # PROP BASE Use_Debug_Libraries 0
45  # PROP BASE Output_Dir "Release"
        # PROP BASE Intermediate_Dir "Release"
```



```
# PROP BASE Target_Dir ""
# PROP Use_MFC 0
# PROP Use_Debug_Libraries 0
# PROP Output_Dir "Release"
5 # PROP Intermediate_Dir "Release"
# PROP Target_Dir ""
OUTDIR=.\\Release
INTDIR=.\\Release

10 ALL : "$(OUTDIR)\\Line21.dll"

CLEAN :
    -@erase ".\\Release\\Line21.dll"
    -@erase ".\\Release\\XDSROM.obj"
15    -@erase ".\\Release\\stable.obj"
    -@erase ".\\Release\\L21Dec.obj"
    -@erase ".\\Release\\XDSMessages.obj"
    -@erase ".\\Release\\Tests.obj"
    -@erase ".\\Release\\funct.obj"
20    -@erase ".\\Release\\Parity.obj"
    -@erase ".\\Release\\Line21.lib"
    -@erase ".\\Release\\Line21.exp"

    "$(OUTDIR)" :
25    if not exist "$(OUTDIR)\\$(NULL)" mkdir "$(OUTDIR)"

# ADD BASE CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D
# _WINDOWS" /YX /c
# ADD CPP /nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D "_WINDOWS" /YX
30 /c
CPP_PROJ=/nologo /MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D "_WINDOWS"\\
/Fp"$(INTDIR)\\Line21.pch" /YX /Fo"$(INTDIR)\\/" /c
CPP_OBJS=.\\Release\\
CPP_SBRS=
35 # ADD BASE MTL /nologo /D "NDEBUG" /win32
# ADD MTL /nologo /D "NDEBUG" /win32
MTL_PROJ=/nologo /D "NDEBUG" /win32
# ADD BASE RSC /1 0x409 /d "NDEBUG"
# ADD RSC /1 0x409 /d "NDEBUG"
40 BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
BSC32_FLAGS=/nologo /o"$(OUTDIR)\\Line21.bsc"
BSC32_SBRS=
45 LINK32=link.exe
```

```
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib odbccp32.lib /nologo
/subsystem:windows /dll /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib advapi32.lib
5 shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib odbccp32.lib /nologo
/subsystem:windows /dll /machine:I386
LINK32_FLAGS=kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib\
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib\
odbccp32.lib /nologo /subsystem:windows /dll /incremental:no\
10 /pdb:"$(OUTDIR)/Line21.pdb" /machine:I386 /def:".L21Dec.def"\
/out:"$(OUTDIR)/Line21.dll" /implib:"$(OUTDIR)/Line21.lib"
DEF_FILE= \
    ".L21Dec.def"
LINK32_OBJS= \
15     "$(INTDIR)/XDSROM.obj" \
        "$(INTDIR)/stable.obj" \
        "$(INTDIR)/L21Dec.obj" \
        "$(INTDIR)/XDSMessages.obj" \
        "$(INTDIR)/Tests.obj" \
20     "$(INTDIR)/funct.obj" \
        "$(INTDIR)/Parity.obj"

"$(OUTDIR)\Line21.dll" : "$(OUTDIR)" $(DEF_FILE) $(LINK32_OBJS)
    $(LINK32) @<<
25     $(LINK32_FLAGS) $(LINK32_OBJS)
    <<

!ELSEIF "$(CFG)" == "Line21 - Win32 Debug"

30     # PROP BASE Use_MFC 0
    # PROP BASE Use_Debug_Libraries 1
    # PROP BASE Output_Dir "Debug"
    # PROP BASE Intermediate_Dir "Debug"
    # PROP BASE Target_Dir ""
35     # PROP Use_MFC 0
    # PROP Use_Debug_Libraries 1
    # PROP Output_Dir "Debug"
    # PROP Intermediate_Dir "Debug"
    # PROP Target_Dir ""
40     OUTDIR=.\\Debug
    INTDIR=.\\Debug

    ALL : "$(OUTDIR)\Line21.dll"

45     CLEAN :
        -@erase ".\\Debug\\vc40.pdb"
```

```

-@erase ".\Debug\vc40.idb"
-@erase ".\Debug\Line21.dll"
-@erase ".\Debug\stable.obj"
-@erase ".\Debug\L21Dec.obj"
5  -@erase ".\Debug\XDSMessages.obj"
-@erase ".\Debug\Tests.obj"
-@erase ".\Debug\Parity.obj"
-@erase ".\Debug\funct.obj"
-@erase ".\Debug\XDSROM.obj"
10 -@erase ".\Debug\Line21.ilk"
-@erase ".\Debug\Line21.lib"
-@erase ".\Debug\Line21.exp"
-@erase ".\Debug\Line21.pdb"

15 "$(OUTDIR)" :
    if not exist "$(OUTDIR)/$(NULL)" mkdir "$(OUTDIR)"

# ADD BASE CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG" /D
# "_WINDOWS" /YX /c
20 # ADD CPP /nologo /MTd /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG" /D
# "_WINDOWS" /YX /c
CPP_PROJ=/nologo /MTd /W3 /Gm /GX /Zi /Od /D "WIN32" /D "_DEBUG" /D
# "_WINDOWS" \
/Fp"$(INTDIR)/Line21.pch" /YX /Fo"$(INTDIR)/" /Fd"$(INTDIR)/" /c
25 CPP_OBJS=.\\Debug/
CPP_SBRS=
# ADD BASE MTL /nologo /D "_DEBUG" /win32
# ADD MTL /nologo /D "_DEBUG" /win32
MTL_PROJ=/nologo /D "_DEBUG" /win32
30 # ADD BASE RSC /I 0x409 /d "_DEBUG"
# ADD RSC /I 0x409 /d "_DEBUG"
BSC32=bscmake.exe
# ADD BASE BSC32 /nologo
# ADD BSC32 /nologo
35 BSC32_FLAGS=/nologo /o"$(OUTDIR)/Line21.bsc"
BSC32_SBRS=
LINK32=link.exe
# ADD BASE LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib /nologo
40 /subsystem:windows /dll /debug /machine:I386
# ADD LINK32 kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib advapi32.lib
shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib /nologo
/subsystem:windows /dll /debug /machine:I386
LINK32_FLAGS=kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib\
45 advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib\
odbccp32.lib /nologo /subsystem:windows /dll /incremental:yes\

```

```

/pdb:"$(OUTDIR)/Line21.pdb" /debug /machine:I386 /def:".L21Dec.def"\
/out:"$(OUTDIR)/Line21.dll" /implib:"$(OUTDIR)/Line21.lib"
DEF_FILE= \
    ".L21Dec.def"
5  LINK32_OBJS= \
    "$(INTDIR)/stable.obj" \
    "$(INTDIR)/L21Dec.obj" \
    "$(INTDIR)/XDSMessages.obj" \
    "$(INTDIR)/Tests.obj" \
10    "$(INTDIR)/Parity.obj" \
    "$(INTDIR)/funct.obj" \
    "$(INTDIR)/XDSROM.obj"

    "$(OUTDIR)/Line21.dll" : "$(OUTDIR)" $(DEF_FILE) $(LINK32_OBJS)
15    $(LINK32) @<<
    $(LINK32_FLAGS) $(LINK32_OBJS)
    <<

!ENDIF
20
    .c{$(CPP_OBJS)}.obj:
        $(CPP) $(CPP_PROJ) $<

    .cpp{$(CPP_OBJS)}.obj:
25    $(CPP) $(CPP_PROJ) $<

    .cxx{$(CPP_OBJS)}.obj:
        $(CPP) $(CPP_PROJ) $<

30    .c{$(CPP_SBRs)}.sbr:
        $(CPP) $(CPP_PROJ) $<

    .cpp{$(CPP_SBRs)}.sbr:
        $(CPP) $(CPP_PROJ) $<
35    .cxx{$(CPP_SBRs)}.sbr:
        $(CPP) $(CPP_PROJ) $<

#####
40    #####
    # Begin Target

    # Name "Line21 - Win32 Release"
    # Name "Line21 - Win32 Debug"
45    !IF "$(CFG)" == "Line21 - Win32 Release"

```

```
!ELSEIF "$(CFG)" == "Line21 - Win32 Debug"

!ENDIF

5 #####
#####
# Begin Source File

10 SOURCE=.\Tests.c
   DEP_CPP_TESTS=\
       ".\Smachine.h"\
       ".\L21Dec.h"\

15 "$(INTDIR)\Tests.obj" : $(SOURCE) $(DEP_CPP_TESTS) "$(INTDIR)"

# End Source File
20 #####
#####
# Begin Source File

   SOURCE=.\L21Dec.c
25 DEP_CPP_L21DE=\
       ".\L21Dec.h"\
       ".\Smachine.h"\

30 "$(INTDIR)\L21Dec.obj" : $(SOURCE) $(DEP_CPP_L21DE) "$(INTDIR)"

# End Source File
#####
35 #####
# Begin Source File

   SOURCE=.\Parity.c

40 "$(INTDIR)\Parity.obj" : $(SOURCE) "$(INTDIR)"

# End Source File
#####
45 #####
# Begin Source File
```

```
SOURCE=.\stable.c
DEP_CPP_STABL=\
    ".\Smachine.h"\
5    ".\L21Dec.h"\

"$(INTDIR)\stable.obj" : $(SOURCE) $(DEP_CPP_STABL) "$(INTDIR)"

10 # End Source File
#####
####
# Begin Source File

15 SOURCE=.\funct.c
DEP_CPP_FUNCT=\
    ".\L21Dec.h"\
    ".\Smachine.h"\
20

"$(INTDIR)\funct.obj" : $(SOURCE) $(DEP_CPP_FUNCT) "$(INTDIR)"

25 # End Source File
#####
####
# Begin Source File

30 SOURCE=.\L21Dec.def

!IF "$(CFG)" == "Line21 - Win32 Release"

!ELSEIF "$(CFG)" == "Line21 - Win32 Debug"
35 !ENDIF

# End Source File
#####
40 ####
# Begin Source File

SOURCE=.\XDSROM.c
DEP_CPP_XDSRO=\
45    ".\L21Dec.h"\
    ".\Smachine.h"
```

"\$(INTDIR)\XDSROM.obj" : \$(SOURCE) \$(DEP\_CPP\_XDSRO) "\$(INTDIR)"

5

# End Source File

#####

#####

# Begin Source File

10

SOURCE=. \XDSMessages.c

DEP\_CPP\_XDSME=\

    ".\L21Dec.h"\

    ".\Smachine.h"\

15

"\$(INTDIR)\XDSMessages.obj" : \$(SOURCE) \$(DEP\_CPP\_XDSME) "\$(INTDIR)"

20

# End Source File

# End Target

# End Project

#####

#####

25

#####

[illegible]



```
/MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D "_WINDOWS" /Fp"Release/Line21.pch"
/YX /Fo"Release/" /c __/nologo /D "NDEBUG" /win32 __/l 0x409 /d "NDEBUG" __/nologo
/o"Release/Line21.bsc" __y0_kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib /nologo
5 /subsystem:windows /dll /incremental:no /pdb:"Release/Line21.pdb" /machine:I386
/def:".\\L21Dec.def" /out:"Release/Line21.dll" /implib:"Release/Line21.lib"
_J:\\Vb60\\Line21\\Tests.c __J:\\Vb60\\Line21\\L21Dec.c __J:\\Vb60\\Line21\\Parity.c __J:\\Vb6
0\\Line21\\stable.c __J:\\Vb60\\Line21\\funct.c __J:\\Vb60\\Line21\\L21Dec.def __J:\\Vb60\\Line21\\
XDSROM.c __J:\\Vb60\\Line21\\XDSMessages.c __Line21 - Win32 Release
10 Line21.mak __CProject_Line21 - Win32 Release_Line21 - Win32 Release_ê __Line21 -
Win32 Debug_ê __SSBR __CTargetItem __SSBR __.\\Tests.c_
CFileItem __SSBR
.\\L21Dec.c_ CFileItem __SSBR
.\\Parity.c_ CFileItem __SSBR
15 .\\stable.c_ CFileItem __SSBR .\\funct.c_ CFileItem __SSBR.\\L21Dec.def_
CFileItem __SSBR
.\\XDSROM.c_ CFileItem __SSBR.\\XDSMessages.c_
CFileItem __SSBR_dep __CDependencyContainer __SSBR
Smachine.h __CDependencyFile __SSBR_L21Dec.h __CDependencyFile __SSBR __DJW_
20 dep __CDependencyContainer __SSBR
Smachine.h __CDependencyFile __SSBR_L21Dec.h __CDependencyFile __SSBR __DJW_
DJW __DJWE& Ççst¥m-. /0123456789
_m_LINE21 - WIN32 DEBUG __{/nologo /MTd /W3 /Gm /GX /Zi /Od /D "WIN32" /D
" _DEBUG" /D " _WINDOWS" /Fp"Debug/Line21.pch" /YX /Fo"Debug/" /Fd"Debug/" /c
25 __/nologo /D " _DEBUG" /win32 __/l 0x409 /d " _DEBUG" __/nologo /o"Debug/Line21.bsc"
__y2_kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib advapi32.lib shell32.lib
ole32.lib oleaut32.lib uuid.lib odbccp32.lib /nologo /subsystem:windows /dll
/incremental:yes /pdb:"Debug/Line21.pdb" /debug /machine:I386 /def:".\\L21Dec.def"
/out:"Debug/Line21.dll" /implib:"Debug/Line21.lib"
30 __J:\\Vb60\\Line21\\Tests.c __J:\\Vb60\\Line21\\L21Dec.c __J:\\Vb60\\Line21\\Parity.c __J:\\Vb6
0\\Line21\\stable.c __J:\\Vb60\\Line21\\funct.c __J:\\Vb60\\Line21\\L21Dec.def __J:\\Vb60\\Line21\\
XDSROM.c __J:\\Vb60\\Line21\\XDSMessages.c __LINE21 - WIN32 RELEASE __j/nologo
/MT /W3 /GX /O2 /D "WIN32" /D "NDEBUG" /D " _WINDOWS" /Fp"Release/Line21.pch"
/YX /Fo"Release/" /c __/nologo /D "NDEBUG" /win32 __/l 0x409 /d "NDEBUG" __/nologo
35 /o"Release/Line21.bsc" __y0_kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbccp32.lib /nologo
/subsystem:windows /dll /incremental:no /pdb:"Release/Line21.pdb" /machine:I386
/def:".\\L21Dec.def" /out:"Release/Line21.dll" /iE& Ççst¥m; <=>?@ABC
```

Microsoft C/C++ program database 2.00

[illegible]



5

10

15

20

25

30

35

40

45





ú\_@ D  
 \_ß\_@ C  
 \_Î\_@ B  
 \_°\_Ñ@ A  
 5 \_2\_ý@ I  
 \_@ H  
 \_B\_@ J  
 \_ä\_@ Q  
 \_O\_@ K  
 10 \_{ \_@ L  
  
 α\_@ M  
 \_Á\_@ N  
 \_Ö\_@ O  
 15 \_ø\_ \_  
 \_  
 \_“î\_“É\_(( ( ö \ Ž ± ““s\_ ( Ĩ  
 □æ\_ d\_ ) □þ\_ Š\_ !□‡\_ &□Ý\_ +K\_ %0\_ â\_ 5\_ 9Ú\_ >\\_ 'C□\_ H5\_  
 20 \_\$Mò\_ RA\_ #W\_ \\*\_ "an\_ d□P\_ g" \_l□\_ &q¢\_ t-  
 \_y\_ +~z\_ \*fž\_ †|\_ ,\_  
 □ú\_ •Á\_ š□È\_ Ÿ\_ !□\_ ©\_ ®í\_ ³x\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—  
 ¢³Ž\_ “\_—¢³Ž\_ ... ³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ ' \_—¢B\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—  
 ¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ ' [i\_ ' [i\_ “\_—¢³Ž\_ “\_—¢³Ž\_ ... ³Ž\_ “\_—¢³Ž\_ “\_—  
 25 ¢³Ž\_ “\_—¢³Ž\_ ... ³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—¢³Ž\_ “\_—  
 ¢³Ž\_ ... —¢³|\_ □o\_ [ \_O\_ □‘\_ @ \_e\_ ž±\_ Ð  
 \_n\_ Y  
 P\_ T\_ □§\_ x\_ x\_ 2\_™\_ o\_ ' Ç\_ Ô\_ ñ\_ ' —¢³' \_ ' \_Ž\_ □\_ ... —¢³x\_ : \_|\_  
 30 \_o\_ □e\_ Y\_ Ĩ\_ < Ĩ\_ ; æ\_ X\_ æ\_ \$\_ æ\_ 3\_ æ\_ 1\_ □h\_ ü\_ Ñ\_ n\_ žÑ\_ %\_ Ñ\_ E  
 Ñ\_ M\_ Ñ\_ = Ñ\_ ` Ñ\_ Y\_ □þ\_ %\_ þ\_ m\_ þ\_ n\_ e\_ žÝ\_ :\_ Ý\_ m\_ Ý\_  
 ; žÁ\_ \$\_ Á\_ h\_ Á\_ g\_ "Á\_ \$žÁ\_ "&Á\_ ü!(Á\_ .\*žÁ\_ i, □Á\_ l\_ Á\_ #0Á\_ U2žÁ\_ 04Á\_  
 6Á\_ L8□Á\_ 1:Á\_ k<žÁ\_ 2>Á\_ D@Á\_ jBÁ\_ 3DÁ\_ XFžÁ\_ WHÁ\_ 4J□Á\_ <L□Á\_  
 mNÁ\_ 5PÁ\_ ;RA\_ /TA\_ 6VÁ\_ VXÁ\_ 7Z□Á\_ :Á\_ 9^Á\_ 8' \_ 6aò\_ 4bò\_ 9c\_  
 35 5dž¶\_ T¶\_ h□¶\_ -  
 j□¶\_ l¶\_ fn¶\_ Kp□¶\_ Cr¼\_ ;t¼\_ 7v¼\_ 8x¼\_ <z□¼\_ D|¼\_ 3~¼\_ 6€¼\_ : \_□¼\_ 2,,ž  
 ¼\_ 4†¼\_ L^¼\_ 5Š¼\_ /Œ¼\_ 1Ž¼\_ .□¼\_ U'¼\_ "¼\_ V-  
 ¼\_ k'¼\_ 0š¼\_ Wæ□¼\_ !ž¼\_ 9¼\_ X¢¼\_ hœž¼\_ j|¼\_ i"□¼\_ \$¹¼\_ m~□¼\_ l@¼\_ #  
 °ž¼\_ "²¼\_ g'¼\_ ¶" \_ 8ž©\_ - 7|\_ ø!|\_ i>|\_ 1¼□|\_ j½|\_ W¾|\_ V¿□|\_  
 40 0Ä|\_ "Á□\_ MÄf\_ ÄÄ\_ YÄ□\_ ÄÈ\_ `Æ□□\_  
 Ç\_  
 ÈÈžu\_ ÉÆ\_ Æ\_ Æ\_ Æ\_ Æ\_ Æ\_ o\_ ×\_ ©\_ ×\_ e\_ ×\_ □\_ ×\_ □\_ ×\_ e\_ ×\_ e\_ ×\_ □\_ È\_ Æ\_ È\_ o\_ Æ\_ Æ\_  
 Æ\_ "h\_ “©\_ “©\_ “e\_ “f\_ “u\_ “h\_ “u\_ “h\_ “□\_ “□\_ “u\_ “e\_ “f\_ “f\_ “□\_ “□\_ “u\_ “f\_ “h\_ “h\_ “f\_ “e\_ “e\_  
 “©\_ “e\_ “e\_ “e\_ “e\_ “u\_ “©\_ “e\_ “e\_ “e\_ Æ\_ Æ\_ Æ\_ Æ\_ “□\_ “e\_ “h\_ “□\_ “©\_ “f\_ “u\_ “s\_ “z\_ “\_ “d\_ “  
 45 Ů\_ “5\_ “\\_ “\_ “A\_ “Ů\_ \* \_ “ã\_ “u\_ “Ů\_ “\_ “ã\_ “i\_ “\_ “i\_ “u\_ “i\_ “K\_ “\_ “ã\_ “u\_ “i\_ “\_ “\_ “\_ “u\_ “Ů\_ “Ů\_ Æ\_ o\_  
 Æ\_ Æ\_ Æ\_ Æ\_ Æ\_ Æ\_ Æ\_ Æ\_ Æ\_ È\_ o\_ È\_ o\_ È\_ o\_ È\_ o\_ | \_o\_ □e\_ ö\_ Y\_ —\_







```

_ _ _ _ _ " _ _ _ _ _ 10_{I^TM9'/_names/ncb/targetinfo/ncb/moduleinfo/ncb/module/J:\Vb60\Line2
1\Tests.c/ncb/module/J:\Vb60\Line21\Smachine.h/ncb/module/J:\Vb60\Line21\L21Dec.h/nc
b/module/J:\Vb60\Line21\L21Dec.c/ncb/module/J:\Vb60\Line21\Parity.c/ncb/module/J:\Vb6
0\Line21\stable.c/ncb/module/J:\Vb60\Line21\funct.c/ncb/module/J:\Vb60\Line21\XDSROM
.c/ncb/module/J:\Vb60\Line21\XDMessages.c/ncb/target/Line21 - Win32
Debug/ncb/target/LINE21 - WIN32 RELEASE_ _ , ? ' _ _ "

```

\_\_\_\_\_ ' \_\_\_\_\_ p \_\_\_\_\_, \_\_\_\_\_ G \_\_\_\_\_ p # \_\_\_\_\_

10 Èÿž\_ \_□\\_üõž□\_,÷žx\_0ûžL\_8÷ž(,øž\_ÈúžÔ

öžĐ\_ \_äÈ\_ \_4üž\_ \_ÿž\_ Øÿž\_ '\$%&!' "# \_\_\_\_\_

File: Line21Def.bas

```
Attribute VB_Name = "Line21Def"
5 Public Const APPNAME As String = "Line21Watch"
  Public Const SETTING As String = "Settings"
  Public Const TCOMF As String = "COMF"
  Public Const IP As String = "IP"
  Public Const PORT As String = "Port"
10 Option Explicit

  Declare Function Line21Decode Lib "Line21.dll" _
    (ByVal field As Long, ByVal ch As Byte, ByVal mes As String) _
15   As Long

  Declare Function ShiftRight Lib "Line21.dll" _
    (ByVal Val As Long, ByVal Shift As Long) As Long

20  Declare Function ShiftLeft Lib "Line21.dll" _
    (ByVal Val As Long, ByVal Shift As Long) As Long
```

File: Line21Watch.vbp

Type=Exe

```
5 Reference=*\G{00020430-0000-0000-C000-
000000000046}#2.0#0#C:\WINDOWS\SYSTEM\StdOle2.tlb#OLE Automation
Object={648A5603-2C6E-101B-82B6-000000000014}#1.1#0; MSCOMM32.OCX
Object={248DD890-BB45-11CF-9ABC-0080C7E7B78D}#1.0#0; MSWINSCK.OCX
Form=FrmLine21Watch.frm
10 Module=Line21Def; Line21Def.bas
Startup="FrmLine21Watch"
HelpFile=""
Command32=""
Name="Line21 Watch"
15 HelpContextID="0"
CompatibleMode="0"
MajorVer=1
MinorVer=0
RevisionVer=0
20 AutoIncrementVer=0
ServerSupportFiles=0
VersionCompanyName="norpak"
CompilationType=0
OptimizationType=0
25 FavorPentiumPro(tm)=0
CodeViewDebugInfo=0
NoAliasing=0
BoundsCheck=0
OverflowCheck=0
30 FIPointCheck=0
FDIVCheck=0
UnroundedFP=0
StartMode=0
Unattended=0
35 Retained=0
ThreadPerObject=0
MaxNumberOfThreads=1
```

File: Line21Watch.vbw

FrmLine21Watch = 115, 120, 649, 565, C, 46, 48, 580, 493, C

5 Line21Def = 69, 72, 603, 517, C

File: Line21Watch.vbw

File: Parity.c

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)

5 All rights reserved

```
unsigned char parity[256] = {
10  0x80, 0x01, 0x02, 0x83, 0x04, 0x85, 0x86, 0x07,
    0x08, 0x89, 0x8a, 0x0b, 0x8c, 0x0d, 0x0e, 0x8f,
    0x10, 0x91, 0x92, 0x13, 0x94, 0x15, 0x16, 0x97,
    0x98, 0x19, 0x1a, 0x9b, 0x1c, 0x9d, 0x9e, 0x1f,
    0x20, 0xa1, 0xa2, 0x23, 0xa4, 0x25, 0x26, 0xa7,
15  0xa8, 0x29, 0x2a, 0xab, 0x2c, 0xad, 0xae, 0x2f,
    0xb0, 0x31, 0x32, 0xb3, 0x34, 0xb5, 0xb6, 0x37,
    0x38, 0xb9, 0xba, 0xb3, 0xbc, 0x3d, 0x3e, 0xbf,
    0x40, 0xc1, 0xc2, 0x43, 0xc4, 0x45, 0x46, 0xc7,
    0xc8, 0x49, 0x4a, 0xcb, 0x4c, 0xcd, 0xce, 0x4f,
20  0xd0, 0x51, 0x52, 0xd3, 0x54, 0xd5, 0xd6, 0x57,
    0x58, 0xd9, 0xda, 0x5b, 0xdc, 0x5d, 0x5e, 0xdf,
    0xe0, 0x61, 0x62, 0xe3, 0x64, 0xe5, 0xe6, 0x67,
    0x68, 0xe9, 0xea, 0x6b, 0xec, 0x6d, 0x6e, 0xef,
    0x70, 0xf1, 0xf2, 0x73, 0xf4, 0x75, 0x76, 0xf7,
25  0xf8, 0x79, 0x7a, 0xfb, 0x7c, 0xfd, 0xfe, 0x7f,
    0x80, 0x01, 0x02, 0x83, 0x04, 0x85, 0x86, 0x07,
    0x08, 0x89, 0x8a, 0x0b, 0x8c, 0x0d, 0x0e, 0x8f,
    0x10, 0x91, 0x92, 0x13, 0x94, 0x15, 0x16, 0x97,
    0x98, 0x19, 0x1a, 0x9b, 0x1c, 0x9d, 0x9e, 0x1f,
30  0x20, 0xa1, 0xa2, 0x23, 0xa4, 0x25, 0x26, 0xa7,
    0xa8, 0x29, 0x2a, 0xab, 0x2c, 0xad, 0xae, 0x2f,
    0xb0, 0x31, 0x32, 0xb3, 0x34, 0xb5, 0xb6, 0x37,
    0x38, 0xb9, 0xba, 0xb3, 0xbc, 0x3d, 0x3e, 0xbf,
    0x40, 0xc1, 0xc2, 0x43, 0xc4, 0x45, 0x46, 0xc7,
35  0xc8, 0x49, 0x4a, 0xcb, 0x4c, 0xcd, 0xce, 0x4f,
    0xd0, 0x51, 0x52, 0xd3, 0x54, 0xd5, 0xd6, 0x57,
    0x58, 0xd9, 0xda, 0x5b, 0xdc, 0x5d, 0x5e, 0xdf,
    0xe0, 0x61, 0x62, 0xe3, 0x64, 0xe5, 0xe6, 0x67,
    0x68, 0xe9, 0xea, 0x6b, 0xec, 0x6d, 0x6e, 0xef,
40  0x70, 0xf1, 0xf2, 0x73, 0xf4, 0x75, 0x76, 0xf7,
    0xf8, 0x79, 0x7a, 0xfb, 0x7c, 0xfd, 0xfe, 0x7f};
```

File: Smachine.h

```
#define WINKCLASS 0x42
5  #define WINKSUMSTART WINKCLASS + 0x0D
   #define XDSEND 0x0F

   #define OVERWRITE " Error: Overwrite"
   #define CONTNOSTART " Error: Continue with no Start"
10  #define TOOLONG " Error: too many bytes"
   #define BADCHECKSUM " Error: bad check sum"
   #define TOOSHORT " Error: too few bytes"

   #define XDSSTART "XDS "
15  #define C1START "C1 "
   #define C2START "C2 "
   #define T1START "T1 "
   #define T2START "T2 "
   #define C3START "C3 "
20  #define C4START "C4 "
   #define T3START "T3 "
   #define T4START "T4 "
   #define ONESP " "
   #define TWOSP " "
25  #define SEVENSP " "
   #define MPAA " MPAA - "
   #define TVPG " TVPG - "
   #define OTHER " OTHER - non-USA"
   #define TVPG_FV ",FV"
30  #define TVPG_V ",V"
   #define TVPG_S ",S"
   #define TVPG_L ",L"
   #define TVPG_D ",D"

35  #define TRUE 1
   #define FALSE 0

   #define C1 0
   #define C2 1
40  #define CAP 0
   #define TXT 1
   #define FIELD1 1
   #define FIELD2 2
   #define POPUP 0
45  #define PAINTON 1
   #define ROLLUP2 2
```

```
#define ROLLUP3 3
#define ROLLUP4 4
#define T1 1
#define T2 2
5

typedef int (__cdecl *FPTR)(), *SPTR;

typedef union sval{
10   FPTR fptr;
   SPTR dptr;
}SVAL;

#define CLASSMAX 7
15 #define RECORDMAX 128
#define MESLEN 6

#define XDSSWink 0
#define XDSCVChip 1
20 #define XDSCMax 2

struct XDSCState {
   BOOL Open;
   int Length;
25   char Data[33];
   int CheckSum;
};

struct cState {
30   int CurrentCtrl[2];           // Save control character which starts pair
   int Channel[2];              // Channel 1 or 2 (control high or low)
   int Mode[2];                 // Mode captioning or text
   char Text[2][2][2][256];     // The saved text
   int TextLength[2][2][2];     // The length of the saved text
35   int Type[2][2];              // The type of caption
   int XDSCClass;               // The current XDSC class
   int XDSCRecord;              // The current XDSC record
   struct XDSCState XDSC[CLASSMAX][RECORDMAX]; // stored data for current XDSC state
   SVAL *NextState[2];          // pointer to the next state
40 };

#define STATE(x) SVAL x##[] = {
#define TEST(x) &##x##,
#define NEXT(x) NULL, ##x##,
45 #define DEFAULT &DefaultTest,
#define DNEXT(x) NULL, ##x##};
```

File: stable.c

```
#include <windows.h>
5  #include <stdio.h>
    #include "sMachine.h"
    #include "L21Dec.h"

    extern SVAL DataState[];
10  extern SVAL ControlState[];
    extern SVAL XDSSstartState[];
    extern SVAL XDSContState[];
    extern SVAL XDSDataState[];
    extern SVAL XDSEndState[];
15  extern SVAL URLDataState[];

    #pragma warning(disable : 4047)
    /*
    *   State Tables
20  *
    *   This is what drives the state machine. Each state has one or more possible
    *   sub-state changes. Each of these is made up of a test followed by a list
    *   of functions to execute if the test passes and a new state to go to. The
    *   state machine tries each test in order. The first test that passes has its
25  *   functions executed and the state changed to its next state. The state
    *   machine then exits without trying any more states. The last test on the list
    *   for a state must be the default test which always returns true.
    */

30  // State DataState
    // This state is to go when we have text data
    //

    STATE(DataState)
35  TEST(NullTest) NEXT(DataState)
    TEST(F1XDSTest) NEXT(DataState)
    TEST(XDSSstartTest) &XDSSBegin, NEXT(XDSSstartState)
    TEST(XDSContTest) &XDSContBegin, NEXT(XDSContState)
    TEST(XDSEndTest) NEXT(XDSEndState)
40  TEST(ControlTest) &ControlSave, NEXT(ControlState)
    DEFAULT &DataSave, DNEXT(DataState)

    // State ControlState
    // In this state we have received a control character and the next
45  // character will decide what action we should take. Warning the control
    // pair may be repeated but in this case we do not care. Normally you
```



```
// require extra states to filter out repeats.

STATE(ControlState)
  TEST(NullTest) NEXT(ControlState)
5  TEST(F1XDSTest) NEXT(DataState)
  TEST(XDSSStartTest) &XDSTBegin, NEXT(XDSSStartState)
  TEST(XDSContTest) &XDSTBegin, NEXT(XDSContState)
  TEST(XDSEndTest) NEXT(XDSEndState)
  TEST(ControlTest) &ControlSave, NEXT(ControlState)
10 TEST(PopOnTest) &PopOnDecode, NEXT(DataState)
  TEST(RollUpTest) &RollUpDecode, NEXT(DataState)
  TEST(PaintOnTest) &PaintOnDecode, NEXT(DataState)
  TEST(URLTest) &URLDecode, NEXT(URLDataState)
  TEST(TextModeTest) &TextDecode, NEXT(DataState)
15 TEST(EndOfCaptionTest) &PopOnDecode, NEXT(DataState)
  TEST(PACTest) &DumpText, NEXT(DataState)
  TEST(CRTest) &ChannelChange, &DumpText, NEXT(DataState)
  TEST(CtrlChangeTest) &ChannelChange, NEXT(DataState)
  DEFAULT DNEXT(DataState)
20
  // State XDSSStartState
  // In this case we have received a an XDS Undefined Start character (0D)
  // The next character determines which class function to execute.

25 STATE(XDSSStartState)
  TEST(NullTest) NEXT(XDSSStartState)
  DEFAULT &XDSTStart, DNEXT(XDSDDataState)

  // State XDSContState
30 // In this case we have received a an XDS Undefined Continue character (0E)
  // The next character determines which class function to execute.

  STATE(XDSContState)
  TEST(NullTest) NEXT(XDSContState)
35  DEFAULT &XDSTCont, DNEXT(XDSDDataState)

  // State XDSDDataState
  // In this state we have received the first character of
  // the XDS stream. Want to stay in this state until
40 // we have receive the full XDS Message.

  STATE(XDSDDataState)
  TEST(XDSEndTest) NEXT(XDSEndState)
  TEST(XDSSStartTest) &XDSTBegin, NEXT(XDSSStartState)
45 TEST(XDSContTest) &XDSTBegin, NEXT(XDSContState)
  TEST(ControlTest) &ControlSave, NEXT(ControlState)
```



File: Tests.c

```
5  #include <windows.h>
   #include <stdio.h>
   #include "sMachine.h"
   #include "L21Dec.h"

10 int DefaultTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
   {
       return(TRUE);
   }

15 int NullTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
   {
       if(ch == 0x00)
           return(TRUE);
       else
20     return(FALSE);
   }

   int FIXDSTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
   {
25     if((ch < 0x10) && (field == FIELD1))
           return(TRUE);
       else
           return(FALSE);
   }

30 int XDSTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
   {
       if((ch > 0) && (ch < 0x0f) && ((ch & 1) == 1))
           return(TRUE);
35     else
           return(FALSE);
   }

   int XDSTestCont(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
40     {
       if((ch > 0) && (ch < 0x0f) && ((ch & 1) == 0))
           return(TRUE);
       else
           return(FALSE);
45     }
```

```
int XDSEndTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    if(ch == 0x0f)
        return(TRUE);
5    else
        return(FALSE);
}

10 int ControlTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    if((ch > 0x0f) && (ch < 0x20))
        return(TRUE);
    else
15    return(FALSE);
}

int TextModeTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
20    int c;

    c = cStatePtr->CurrentCtrl[field - 1];
    if((c == 0x14) || (c == 0x15) || (c == 0x1c) || (c == 0x1d))
    {
25        if((ch == 0x2a) || (ch == 0x2b))
            return(TRUE);
        }
    return(FALSE);
}

30 int RollUpTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    int c;

35    c = cStatePtr->CurrentCtrl[field - 1];
    if((c == 0x14) || (c == 0x15) || (c == 0x1c) || (c == 0x1d))
    {
        if((ch > 0x24) && (ch < 0x28))
            return(TRUE);
40    }
    return(FALSE);
}

int PaintOnTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
45 {
    int c;
```

```
c = cStatePtr->CurrentCtrl[field - 1];
if((c == 0x14) || (c == 0x15) || (c == 0x1c) || (c == 0x1d))
{
5   if(ch == 0x29)
      return(TRUE);
}
return(FALSE);
}

10 int PopOnTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    int c;

15   c = cStatePtr->CurrentCtrl[field - 1];
    if((c == 0x14) || (c == 0x15) || (c == 0x1c) || (c == 0x1d))
    {
        if(ch == 0x20)
            return(TRUE);
20    }
    return(FALSE);
}

int EndOfCaptionTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
25 {
    int c;

    c = cStatePtr->CurrentCtrl[field - 1];
    if((c == 0x14) || (c == 0x15) || (c == 0x1c) || (c == 0x1d))
30    {
        if(ch == 0x2f)
            return(TRUE);
        }
    return(FALSE);
35 }

int PACTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
{
    int c;

40   c = cStatePtr->CurrentCtrl[field - 1];
    if((c > 0xf) && (c < 0x20))
    {
        if(ch > 0x3f)
45     return(TRUE);
    }
}
```

```
    return(FALSE);
}

int CtrlChangeTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
5  {
    int f, c, ctrl;

    f = field - 1;
    c = cStatePtr->Channel[f];
10  ctrl = cStatePtr->CurrentCtrl[f];
    if(((ch > 0x1f) && (ch < 0x40)) && (((ctrl < 0x18) && (c != C1)) || ((ctrl > 0x17) && (c !=
C2))))
        return(TRUE);
    else
15  return(FALSE);
}

int CRTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
20  {
    int c;

    c = cStatePtr->CurrentCtrl[field - 1];
    if((c == 0x14) || (c == 0x15) || (c == 0x1c) || (c == 0x1d))
    {
25  if(ch == 0x2d)
        return(TRUE);
    }
    return(FALSE);
}

30  int URLTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
    {
        int c;

35  c = cStatePtr->CurrentCtrl[field - 1];
        if((field == FIELD1) && (c == 0x1c))
        {
            if((ch == 0x2a) || (ch == 0x2b))
                return(TRUE);
40  }
        return(FALSE);
    }

int URLEndTest(long field, unsigned char ch, char mes[], struct cState *cStatePtr)
45  {
    int f, l;
```

```
char *t;

f = field - 1;
l = cStatePtr->TextLength[f][C2][TXT];
5  t = cStatePtr->Text[f][C2][TXT];
    if(l > 5)
        if((ch == ']') && (t[l-5] == '['))
            {
                if(((t[l-4] >= '0') && (t[l-4] <= '9')) || ((t[l-4] >= 'A') && (t[l-4] <= 'F')) || ((t[l-4] >= 'a')
10  && (t[l-4] <= 'f'))
                    if(((t[l-3] >= '0') && (t[l-3] <= '9')) || ((t[l-3] >= 'A') && (t[l-3] <= 'F')) || ((t[l-3] >= 'a')
&& (t[l-3] <= 'f'))
                        if(((t[l-2] >= '0') && (t[l-2] <= '9')) || ((t[l-2] >= 'A') && (t[l-2] <= 'F')) || ((t[l-2] >= 'a')
&& (t[l-2] <= 'f'))
15  if(((t[l-1] >= '0') && (t[l-1] <= '9')) || ((t[l-1] >= 'A') && (t[l-1] <= 'F')) || ((t[l-1] >=
'a') && (t[l-1] <= 'f'))
                            return(TRUE);
                        }
                    return(FALSE);
20  }
```

File: XDSMessages.c

```
#include <windows.h>
#include <stdio.h>
5  #include <string.h>
    #include "L21Dec.h"
    #include "sMachine.h"

extern char XDSMes[CLASSMAX][RECORDMAX][MESLEN];
10 extern char *mpaaRatings[];
    extern char *tvpgratings[];
    extern char flagmask[];
    extern char *types[];

15 void Def(char mes[], struct cState *cStatePtr)
    {
        strcpy(mes,XDSSTART);
        strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
        strcat(mes,TWOSP);
20  cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Data[cStatePtr-
    >XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Length] = 0;
        strcat(mes, cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Data);
    }

25 void Vcp(char mes[], struct cState *cStatePtr)
    {
        char b1, b2, type;
        char flags;

30  strcpy(mes,XDSSTART);
        strcat(mes,XDSMes[cStatePtr->XDSClass][cStatePtr->XDSRecord]);
        b1 = cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Data[0];
        b2 = cStatePtr->XDS[cStatePtr->XDSClass][cStatePtr->XDSRecord].Data[1];
        type = ((b1 >> 3) & 3);
35  if((type == 0) || (type == 2)) //MPAA
        {
            strcat(mes, MPAA);
            strcat(mes, mpaaRatings[b1 & 7]);
        }
40  else if(type == 1) // TVPG
        {
            strcat(mes, TVPG);
            strcat(mes, tvpgRatings[b2 & 7]);
            flags = (b1 >> 5) & 1;
45  flags = flags | ((b2 & 0x38) >> 2);
            flags = flags | ((b2 & 0x20) >> 1);
```



```

    flags = flags & flagmask[b2 & 7];
    if(flags & 0x10)
        strcat(mes,TVPG_FV);
    if(flags & 8)
5       strcat(mes,TVPG_V);
    if(flags & 4)
        strcat(mes,TVPG_S);
    if(flags & 2)
        strcat(mes,TVPG_L);
10     if(flags & 1)
        strcat(mes,TVPG_D);
    }
    else // Other
    {
15     strcat(mes, OTHER);
    }
}

void Typ(char mes[], struct cState *cStatePtr)
20 {
    int i;

    strcpy(mes,XDSSTART);
    strcat(mes,XDSMes[cStatePtr->XDSCClass][cStatePtr->XDSRecord]);
25     strcat(mes,TWOSP);
    for(i = 0;i < cStatePtr->XDS[cStatePtr->XDSCClass][cStatePtr->XDSRecord].Length;i++)
    {
        strcat(mes, types[cStatePtr->XDS[cStatePtr->XDSCClass][cStatePtr->XDSRecord].Data[i]
- 0x20]);
30     strcat(mes, ONESP);
    }
}
```



```

32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
{32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, // Reserved
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
5 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
10 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
{32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, // Undefined Class
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
15 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,
20 char MinLength[CLASSMAX][RECORDMAX] = {
    { 0, 4, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Current Class
      0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
25 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
30 { 0, 4, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Future Class
      0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
35 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
      { 0, 2, 4, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Channel Class
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
40 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
        2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
45 { 0, 6, 6, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Miscellaneous
      2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,

```

```
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
5 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2},
{ 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Public Service
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
10 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2},
15 { 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Reserved
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
20 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2},
{ 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, // Undefined Class
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
25 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
30 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2},
};
```

```
char XDSMes[CLASSMAX][RECORDMAX][MESLEN] = {
{"Null ", "Start", "Durat", "Show ", "Type ", "VChip", "Audio", "Capti", // Current Class
"0108 ", "Aspec", "010A ", "010B ", "Comp1", "Comp2", "010E ", "010F ",
35 "Desc1", "Desc2", "Desc3", "Desc4", "Desc5", "Desc6", "Desc7", "Desc8",
"0118 ", "0119 ", "011A ", "011B ", "011C ", "011D ", "011E ", "011F ",
"0120 ", "0121 ", "0122 ", "0123 ", "0124 ", "0125 ", "0126 ", "0127 ",
"0128 ", "0129 ", "012A ", "012B ", "012C ", "012D ", "012E ", "012F ",
"0130 ", "0131 ", "0132 ", "0133 ", "0134 ", "0135 ", "0136 ", "0137 ",
40 "0138 ", "0139 ", "013A ", "013B ", "013C ", "013D ", "013E ", "013F ",
"0140 ", "0141 ", "0142 ", "0143 ", "0144 ", "0145 ", "0146 ", "0147 ",
"0148 ", "0149 ", "014A ", "014B ", "014C ", "014D ", "014E ", "014F ",
"0150 ", "0151 ", "0152 ", "0153 ", "0154 ", "0155 ", "0156 ", "0157 ",
"0158 ", "0159 ", "015A ", "015B ", "015C ", "015D ", "015E ", "015F ",
45 "0160 ", "0161 ", "0162 ", "0163 ", "0164 ", "0165 ", "0166 ", "0167 ",
"0168 ", "0169 ", "016A ", "016B ", "016C ", "016D ", "016E ", "016F ",
```

```

"0170 ","0171 ","0172 ","0173 ","0174 ","0175 ","0176 ","0177 ",
"0178 ","0179 ","017A ","017B ","017C ","017D ","017E ","017F "},
{"0300 ","0301 ","0302 ","0303 ","0304 ","0305 ","0306 ","0307 ", // Future Class
"0308 ","0309 ","030A ","030B ","030C ","030D ","030E ","030F ",
5  "0310 ","0311 ","0312 ","0313 ","0314 ","0315 ","0316 ","0317 ",
"0318 ","0319 ","031A ","031B ","031C ","031D ","031E ","031F ",
"0320 ","0321 ","0322 ","0323 ","0324 ","0325 ","0326 ","0327 ",
"0328 ","0329 ","032A ","032B ","032C ","032D ","032E ","032F ",
"0330 ","0331 ","0332 ","0333 ","0334 ","0335 ","0336 ","0337 ",
10  "0338 ","0339 ","033A ","033B ","033C ","033D ","033E ","033F ",
"0340 ","0341 ","0342 ","0343 ","0344 ","0345 ","0346 ","0347 ",
"0348 ","0349 ","034A ","034B ","034C ","034D ","034E ","034F ",
"0350 ","0351 ","0352 ","0353 ","0354 ","0355 ","0356 ","0357 ",
"0358 ","0359 ","035A ","035B ","035C ","035D ","035E ","035F ",
15  "0360 ","0361 ","0362 ","0363 ","0364 ","0365 ","0366 ","0367 ",
"0368 ","0369 ","036A ","036B ","036C ","036D ","036E ","036F ",
"0370 ","0371 ","0372 ","0373 ","0374 ","0375 ","0376 ","0377 ",
"0378 ","0379 ","037A ","037B ","037C ","037D ","037E ","037F "},
{"0500 ","Netwk","Affil","Delay","0504 ","0505 ","0506 ","0507 ", // Channel Class
20  "0508 ","0509 ","050A ","050B ","050C ","050D ","050E ","050F ",
"0510 ","0511 ","0512 ","0513 ","0514 ","0515 ","0516 ","0517 ",
"0518 ","0519 ","051A ","051B ","051C ","051D ","051E ","051F ",
"0520 ","0521 ","0522 ","0523 ","0524 ","0525 ","0526 ","0527 ",
"0528 ","0529 ","052A ","052B ","052C ","052D ","052E ","052F ",
25  "0530 ","0531 ","0532 ","0533 ","0534 ","0535 ","0536 ","0537 ",
"0538 ","0539 ","053A ","053B ","053C ","053D ","053E ","053F ",
"0540 ","0541 ","0542 ","0543 ","0544 ","0545 ","0546 ","0547 ",
"0548 ","0549 ","054A ","054B ","054C ","054D ","054E ","054F ",
"0550 ","0551 ","0552 ","0553 ","0554 ","0555 ","0556 ","0557 ",
30  "0558 ","0559 ","055A ","055B ","055C ","055D ","055E ","055F ",
"0560 ","0561 ","0562 ","0563 ","0564 ","0565 ","0566 ","0567 ",
"0568 ","0569 ","056A ","056B ","056C ","056D ","056E ","056F ",
"0570 ","0571 ","0572 ","0573 ","0574 ","0575 ","0576 ","0577 ",
"0578 ","0579 ","057A ","057B ","057C ","057D ","057E ","057F "},
35  {"0700 ","Time ","Impul","Suppl","Zone ","0705 ","0706 ","0707 ", // Miscellaneous
"0708 ","0709 ","070A ","070B ","070C ","070D ","070E ","070F ",
"0710 ","0711 ","0712 ","0713 ","0714 ","0715 ","0716 ","0717 ",
"0718 ","0719 ","071A ","071B ","071C ","071D ","071E ","071F ",
"0720 ","0721 ","0722 ","0723 ","0724 ","0725 ","0726 ","0727 ",
40  "0728 ","0729 ","072A ","072B ","072C ","072D ","072E ","072F ",
"0730 ","0731 ","0732 ","0733 ","0734 ","0735 ","0736 ","0737 ",
"0738 ","0739 ","073A ","073B ","073C ","073D ","073E ","073F ",
"OObNB","0741 ","0742 ","0743 ","0744 ","0745 ","0746 ","0747 ",
"0748 ","0749 ","074A ","074B ","074C ","074D ","074E ","074F ",
45  "0750 ","0751 ","0752 ","0753 ","0754 ","0755 ","0756 ","0757 ",
"0758 ","0759 ","075A ","075B ","075C ","075D ","075E ","075F ",

```





```
{Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, // Public
Service
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
5 Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def},
10 {Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, //
Reserved
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
15 Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def},
{Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, //
20 Undefined Class
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
25 Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def,
Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def, Def};

char *mpaaRatings[] = {"N/A", "G", "PG", "PG-13", "R", "NC-17", "X", "Not Rated"};
30 char *tvpgRatings[] = {"None", "TV-Y", "TV-Y7 ", "TV-G", "TV-PG ", "TV-14 ", "TV-MA
", "None"};
char flagmask[] = { 0, 0, 0x10, 0, 0xf, 0xf, 0xe, 0};

char *types[] = {
35 "Education", "Entertainment", "Movie", "News", "Religious", "Sports", "OTHER",
"Action",
"Advertisement", "Animated", "Anthology", "Automobile", "Awards", "Baseball",
"Basketball", "Bulletin",
"Business", "Classical", "College", "Combat", "Comedy", "Commentary", "Concert",
40 "Consumer",
"Contemporary", "Crime", "Dance", "Documentary", "Drama", "Elementary", "Erotica",
"Exercise",
"Fantasy", "Farm", "Fashion", "Fiction", "Food", "Football", "Foreign", "Fund Raiser",
"Game/Quiz", "Garden", "Golf", "Government", "Health", "High School", "History",
45 "Hobby",
```





ExtractDB

File: hexdec.pl

```
5  #!/usr/bin/perl

    ## hex dec display

    $a=chr(32);
10  print ">". $a. "<". "\n";

    $a=hex(32);
    print ">". $a. "<". "\n";

15  $a=hex(20);

    print "\n";
    #print pack"h2", "32";
20  $a = sprintf "%lx", (32);

    print $a;
```

File: stamp2db.pl

```
5  #!/usr/bin/perl

# this program access mysql DB and reads off all
#or as much data as you want,
#it then generates a web page to view it in raw form.

10 use DBI;

my($dbh, $sth);

15 $dbh = DBI->connect( 'DBI:mysql:caption:localhost:3306', 'root', '' ) || warn "connecting:
$DBI::errstr\n";

$sth = $dbh->prepare("select * from raw order by recordnum");
20 $sth->execute();

# get all from DB

25 while(@row = $sth->fetchrow()) {
#   print @row[2]."\n";
$tim=@row[1];
$_=@row[2];
$I1=0;$I2=0;
30 if(/^(\d)\s(\d+)\s$/) {$fnum=$I1; $I1=$I2;}
elseif(/^(\d)\s(\d+)\s(\d+)\s$/) {$fnum=$I1; $I1=$I2; $I2=$I3;}
#else {print "OPPS".@row[2]."\n";}
else {}

35 if( ($I1 == 20 || $I1 == 28) && $I2 == 44 ) {}

else
{
40   if(1 == ($fnum+0)) {
#print $tim;
if ($I1 >= 0x20 && $I1 < 0x7f) {$out.=chr($I1);}
elseif($I1 != 0) {$a = sprintf "%lx", $I1+0;
45   $out.="[".$a."];"}
if ($I2 >= 0x20 && $I2 < 0x7f) { $out.=chr($I2);}
```

```
        elsif($12 != 0) { $a = sprintf "%lx", $12+0;
        $out.="[".$12."]";}
        #print "\n";
    }
5    } # end while

    if(2 == ($fnum+0)) {
10    if ($11 >= 0x20 && $11 < 0x7f) {$out2.=chr($11);}
        elsif($11 != 0) {$out2.="[".$11."]";}
        if ($12 >= 0x20 && $12 < 0x7f) {$out2.=chr($12);}
        elsif($12 != 0) {$out2.="[".$12."]";}
    }
15

    }

20    # write out web page with data

    open(FILE,"> /usr/www/names/docs/caption_new.html");
    print FILE "<HTML>Strings or repetitive [20], or [28], are not shown<br><br><font
25    color=blue>Field 1</font><br><br>\n";
    $_=$out;

    # do necessary substitution to flag URL
30    s/(<[^<>]*)\>/^[$1]/g;

    s/^[20\]\^x04/g;
    s/([28\]*)^(\^x04*)(x04)/^<font color=red\>$1$2$3\</font>/g;
    s/^x04/^[20\]/g;
35    $out=$_;

    print FILE $out;

    print FILE "<br><br><font color=blue>Field 2</font><br><br>\n";
40    print FILE $out2;

    close(FILE);
```

## File: funct.obj

```

5 L_ylTM9D_.,directve(D_
_.text_l_P'.debug$F_%_TM_H_B.text;f_b_P'.data_ @_0À.data_
@_0À.debug$F_ / _H_B.text;9t
_P'.data!œ
@_0À.debug$F_½
10 İ
_H_B.text,×
Y_P'.debug$F_c_s_H_B.textQ_}İ
_P'.data_d_@_0À.data_|@_0À.data_•_@_0À.debug$F_-½_H_B.text_C_
_P'.debug$F_ß_İ_H_B.textEü_P'.debug$F_>_N_H_B.textIX_
15 P'.debug$F_ı_±_H_B.textI»_P'.debug$F_'_7_H_B.textCA_
_P'.debug$F_,_”_H_B.textIž_P'.debug$F_ç_÷_H_B.text2_
_P'.debug$F_3_C_H_B.text_M_P'.debug$F_]_m_H_B.text7w_
_P'.debug$F_®_¾_H_B.textB_E_
_P'.data_ú_@_0À.data_ÿ_@_0À.data_@_0À.data_
20 @_0À.data_@_0À.data_@_0À.data_@_0À.data_
@_0À.data_%_@_0À.debug$F_*:_H_B-defaultlib:LIBCMT-defaultlib:OLDNAMES
3À<LS_ŠD$H™+ÄÄø_%□H_Ä_<TS_fi_SVW3ÛŠ$U<H_%øšL_Áà_ÄÄà_□@f
¼_P_...~<I$<
,_%MŠ@^E<,H_Áà_,L_□@+À□<M'ÿÿÿÿò®÷Ñ+ù%LS_<÷'ÿÿÿÿý+Àò®O<LS_Áé_ó¥<
25 L$_fá_óç;ÿÿÿÿ+Àò®÷Ñ+ù%LS_<÷'ÿÿÿÿý+Àò®O<LS_Áé_ó¥<L$_fá_óç<,H_Áà_,L_Áà_
_□@Ç,,_P_<,H_Áà_,L_Áà_□@Ç,,_T_<ŠH_□_KÁá_@_ŠL_]Áá_^□_I%o,,_|_,_|[fÄ_Ä<
_A_e_TM_Error: OverwriteXDS
;_____<TS_fi_SVW3ÛŠ$U<H_%øšL_Áà_ÄÄà_□@f¼_P_...ø<I$<
,_%MŠ@^E<,H_Áà_,L_□@+À□<M'ÿÿÿÿò®÷Ñ+ù%LS_<÷'ÿÿÿÿý+Àò®O<LS_Áé_ó¥<
30 L$_fá_óç;ÿÿÿÿ+Àò®÷Ñ+ù%LS_<÷'ÿÿÿÿý+Àò®O<LS_Áé_ó¥<L$_fá_óç<,H_Áà_,L_Áà_
_□@Ç,,_P_<,H_Áà_,L_Áà_□@Ç,,_T_<ŠH_□_KÁá_@_ŠL_Áá_□_I%o,,_|_,_|^[fÄ_Ä<
_A_e_TM_Error: Continue with no
Start;_____<TS_SVW<,H_3ÛÄÄ_ŠL$_L_ŠÛÄÄ_□4@_œ_|²H_Áæ_²L_<ÆÄÄ_¾¶□
<@<,,_T_;&~_Ä^CE8X<,H_Áà_^_L_[ÄÄ_□@_,ÿ,,_T_ÄB!_,____
35 <L$SVW<□H_Áà_□L_Áà_□_@Ç,,
P_<±H_Áæ_±L_<ÆÄÄ_□_@_¾†□<
<”
T_;&Ž“<TS_ı¾%_Š^_Z_<□H_Áà_□L_□@+À□<M'ÿÿÿÿò®÷Ñ+ù_Ü<÷'ÿÿÿÿú+Àò®O<
ÄÄé_ó¥<Ëfá_óç;ÿÿÿÿ+Àò®÷Ñ+ù_Ü<÷'ÿÿÿÿú+Àò®O<ÄÄé_ó¥<Ëfá_óç_,_^[Ä_¾†;Ä_□”<
40 TS_<5_%,2Š@_²B_<□H_Áà_□L_□@+À□<M'ÿÿÿÿò®÷Ñ+ù_Ü<÷'ÿÿÿÿú+Àò®O<ÄÄé_ó¥<
Ëfá_óç;ÿÿÿÿ+Àò®÷Ñ+ù_Ü<÷'ÿÿÿÿú+Àò®O<ÄÄé_ó¥<Ëfá_óç_,_^[Ä3ÄŠD$_†_|fÄ_
,,”<TS_<5_%,2Š@_²B_<□H_Áà_□L_□@+À□<M'ÿÿÿÿò®÷Ñ+ù_Ü<÷'ÿÿÿÿú+Àò®O<ÄÄé_ó¥<
ÄÄé_ó¥<Ëfá_óç;ÿÿÿÿ+Àò®÷Ñ+ù_Ü<÷'ÿÿÿÿú+Àò®O<ÄÄé_ó¥<Ëfá_óç_,_^[Ä<TS_QRÿ_ufÄ_
_^[ÄÄ1_\_a_,,²/ı!_ÿ_’_U_,_a_—_Ö_)_A_0_Error: bad check sum
45 Error: too many bytes Error: too few
bytesQ_&3Ö<D$ŠTS_<L$_%T□ü_,_Ä_6_<L$SVI<t$_□_M_TŽ_□_Ž_Ò_P_□E-

```

```

<=ÿ} Áâ Š$ _Ö\ _ÿ _ ^[ÃE _; <T$_<L$_ □D'üÇ@_f8_ } Ç@_Ç_,Ñ8_ _ÃÇ@_Ç
,,Ñ<_ _ÄI_@_<L$_ VWI<t$ _¿_ _ŽÇ@_f8_ | _¿_ ŠT$_%ox_ €ú%u_ □_ OÇ_,†8_ €ú&u_ □_ OÇ_,
†8_ €ú'u_ □_ OÇ_,†8_ _ ^ÄI_ _E_ <T$_ VJ<L$_ 3Ä_4'‰D' _f>_ } %F_ _Ç_,Ñ8_ ^Ä_ %oF
_ ^‰_,Ñ<_ _ÄÇ_ _J_ <T$_ <D$_ □L□üÇA_ f9_ } ÇA_Ç_,Đ8_ _ÄÇA_Ç_,Đ<_ _ÄI_ O_ <L$_
5 <D$_ □D'üÇ@_f8_ }
Ç@_ _ÄÇ@_ _Ä2_ T_ <L$_ _%A_ %A_ Ä_ Y_ <L$_ ŠD$_ I< r$<?w_ <D$_ □_ ^f8_ }
Ç@_ _ÄÇ@_ _Ä7_ ^_ <T$_ fì_ S□Jÿ<L$_ VWU<t_ <|_ □_ <□_ O□,Ff¼«_ _Ž_ fú_ ...U_ ...
ö_...°...ÿuV<T$_ <
¾‰
10 ŠN_ J_ 'ÿÿÿÿ<_,f8_ <...+Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ éA_ <T$_
<5‰2ŠI_ J_ 'ÿÿÿÿ<_,f8_ <...+Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ éë_...ÿuL<
T$_ ¿_ '‰_ +ÄSI_ J_ 'ÿÿÿÿò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ é_ <T$_ ¿_ <
_‰
'ÿÿÿÿŠ@_ ^B_ +Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ éN_...ö_...°...ÿuV<T$_
15 <5‰2ŠI_ J_ 'ÿÿÿÿ<_,f8_ <...+Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ éí<T$_
<5‰2ŠI_ J_ 'ÿÿÿÿ<_,f8_ <...+Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ é_
...ÿuJ<T$_ ¿_ <
_‰
'ÿÿÿÿŠ@_ ^B_ +Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ ëH<T$_ ¿_ <
_‰
20 'ÿÿÿÿŠ@_ ^B_ +Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ fá_ óŸ<L$_ Ä_ Ä_ Ä_ Ä_
D_ □_ 'ÿÿÿÿ+Äò@÷Ñ+ù‰L$_ <÷'ÿÿÿÿú+Äò@O<L$_ Áé_ óŸ<L$_ fá_ óŸ<L$_ «_ _ _ ^[fÄ_ Ä_
L~_ Q~_ 1_ { _ { _ Ä_ ü_ x_ x_ G_ u_ M_ r_ R_ r_ i_ o_ |_ o_ Á_ ÷_ l_ ü_ l_ P_ u_ V_ i_
[ _ i_ š_ u_ _ f_ Ÿ_ f_ T4_ T3_ C4_ C3_ T2_ _ T1_ C2_ C1_
25 B_ _c_ .filepÿg_ J:\Vb60\Line21\funct.c.drectve_ (.text_
_.debug$F_ .text_ ;
_.data_ .data_ E_ XD$Mes_ .debug$F_ .text_ ; _ XD$Cont_ .data
_!_ ]_ .debug$F
_.text_ , _ XD$Data_ _ _ .debug$F_ .text
30 _ Q_ XD$End
_.data_ i_ .data_ Ô_ .data_ XD$Disp_ ; .debug$F_
_.text_ F_ .debug$F_ .text_ E_ S_ .debug$F_ .text_ I_ ]
_.debug$F_ .text_ l_ j_ .debug$F_ .text_ C_ x_
_.debug$F_ .text_ I_ ‡_ .debug$F_ .text_ 2_ >_ .debug$F_ .text
35 _ $ _ .debug$F!_ .text" _ 7_ ^ _ " _ .debug$F#_ " _ .text$ _ B_ Ä_ $
_.data% _ Ä_ % _ .data& _ ä_ & _ .data' _ ý_ ' _ .data( _ ( _ .data) _ / _ ) _ .data*_ _ H
*_ _ .data+_ _ h_ + _ .data, _ □ _ , _ .data- _ š _ -
_CType_ .debug$F_ _ $ _ ^3_ XD$Begin_ XD$Start??_ C@_ 0BE@FLOP@?5?5Error?3?5?
5Overwrite?$AA@??_ C@_ 04EPNH@XD$5?5$AA@??_ C@_ 0CB@OHLK@?5?5Error?3?
40 5?5Continue?5with?5no?5Start@_ MaxLength??_ C@_ 0BI@LPLM@?5?5Error?3?5?5bad?5
check?5sum?$AA@??_ C@_ 0BJ@ICAC@?5?5Error?3?5?5too?5many?5bytes?$AA@??_ C
@_ 0BI@NPMA@?5?5Error?3?5?5too?5few?5bytes?$AA@_ MinLength_ ControlSave_ Data
Save_PopOnDecode_RollUpDecode_PaintOnDecode_EndOfCaptionDecode_TextDecode_U
RLDecode_ChannelChange_DumpText??_ C@_ 04DILL@T4?5?5$AA@??_ C@_ 04IGNK
45 @T3?5?5$AA@??_ C@_ 04JHCE@C4?5?5$AA@??_ C@_ 04CJEF@C3?5?5$AA@??_ C

```

@\_04NEMM@T2?5?5?\$AA@??\_C@\_07NDP@?5?5?5?5?5?5?\$AA@??\_C@\_04CCPH  
@T1?5?5?\$AA@??\_C@\_04HLFD@C2?5?5?\$AA@??\_C@\_04INGI@C1?5?5?\$AA@

Page 108 of 108

File: L21Dec.obj

```
L__m™9"!directveİ_
5  _text_8_P`.debug$F_@_P_H_B.text"Z_____P`.debug$F_p_€_H_B.textzŠ____
   _P`.debug$F_,<_H_B.text
   F_P`.debug$F_S_c_H_B.text
   m_P`.debug$F_z_Š_H_B-defaultlib:LIBCMT-defaultlib:OLDNAMES -
   export:_Line21Decode@12-export:_ShiftRight@8-export:_ShiftLeft@8,_Â_
10  _<T$SV3ÉW...ÖupÇ_,£P°<_%,T°<_%J<_%J<_3ö,_%J<_%J_°€3ÿ<_fÆ0J%¼3
   _uiHuã3Â_^[Âfú_t_fú_u_ŠL$_<D$_€á□PQÆRèfÃ3Â_^[Â_____#/_8_A_Q
   _a_™_____S_iV3Û<t$WŠ\$_U<¬°L°<|$_<Â<
   fÂ_QWSVÿ_fÄ_...Âu_<ÂfÂ_f8uöfÂ_ëÜf}t_<Â<
   fÂ_QWSVÿ_fÄ_f}uæ<E_<
15  ]_%,,±L°^[Â_"_O_j_z____<D$_ŠL$_ÓèÂ_
   ____<D$_ŠL$_ÓäÂ_
   ____filepÿg_J:\Vb60\Line21\L21Dec.c.directive__1_CCstateX°____.text____
   ____debug$F____.text____"_____,_debug$F____.text_z_7_
   ____debug$F____.text____
20  _E__.debug$F_____text
   _
   _S
   ____debug$F_____
   _cStatePtr_DllMain@12_Line21Decode@12_DataState_stateMachine_ShiftRight@8_Shif
25  tLeft@8
```



File: Line21.dll

**DYNAMIC LINK LIBRARY**

5 **32bit for Windows 95 and Windows NT**

***Technical File Information:***

**Image File Header**

10 \_\_\_\_\_ *Signature:* 00004550  
                  *Machine:* Intel 386  
                  *Number of Sections:* 0005  
                  *Time Date Stamp:* 39996d0e  
15                *Symbols Pointer:* 00000000  
                  *Number of Symbols:* 00000000  
                  *Size of Optional Header* 00e0  
                  *Characteristics:* File is executable (i.e. no unresolved external references).  
  Line numbers stripped from file.  
20                                        Local symbols stripped from file.  
  32 bit word machine.  
  File is a DLL.

25 **Image Optional Header**

                  \_\_\_\_\_ *Magic:* 010b  
                  *Linker Version:* 3.00  
30                *Size of Code:* 00003800  
                  *Size of Initialized Data:* 00012000  
                  *Size of Uninitialized Data:* 00000000  
                  *Address of Entry Point:* 00002680  
                  *Base of Code:* 00001000  
35                *Base of Data:* 00005000  
                  *Image Base:* 10000000  
                  *Section Alignment:* 00001000  
                  *File Alignment:* 00000200  
                  *Operating System Version:* 4.00  
40                *Image Version:* 0.00  
                  *Subsystem Version:* 4.00  
                  *Reserved1:* 00000000  
                  *Size of Image:* 00019000  
                  *Size of Headers:* 00000400  
45                *Checksum:* 00000000  
                  *Subsystem:* Image runs in the Windows GUI subsystem.

DLL Characteristics: 0000  
 Size of Stack Reserve: 00100000  
 Size of Stack Commit: 00001000  
 Size of Heap Reserve: 00100000  
 5 Size of Heap Commit: 00001000  
 Loader Flags: 00000000  
 Size of Data Directory: 00000010  
 Export Directory Virtual Address: 00005060  
 Export Directory Size: 00000073  
 10 Import Directory Virtual Address: 00017000  
 Import Directory Size: 00000028  
 Base Relocation Table  
 Virtual Address: 00018000  
 Base Relocation Table Size: 00000c40

15

### Export Table

20 \_\_\_\_\_ Name: Line21.dll  
 Characteristics: 00000000  
 Time Date Stamp: 39996d0e  
 Version: 0.00  
 Base: 00000001  
 25 Number of Functions: 00000003  
 Number of Names: 00000003

	Ordinal	Entry Point	Name
	0000	00001010	Line21Decode
30	0001	00001150	ShiftLeft
	0002	00001140	ShiftRight

### Import Table

35

KERNEL32.dll

Ordinal	Function Name
---------	---------------

	0092	GetACP
40	0103	GetProcAddress
	00eb	GetModuleHandleA
	0137	GetVersion
	0062	ExitProcess
	00c7	GetCurrentThreadId
45	0222	TlsSetValue
	021f	TlsAlloc

	0220	TlsFree
	01fd	SetLastError
	0221	TlsGetValue
	00e1	GetLastError
5	0155	HeapCreate
	0157	HeapDestroy
	01fa	SetHandleCount
	00dc	GetFileType
	0116	GetStdHandle
10	0114	GetStartupInfoA
	0044	DeleteCriticalSection
	00e9	GetModuleFileNameA
	0098	GetCPIInfo
	009f	GetCommandLineA
15	00f6	GetOEMCP
	008b	FreeEnvironmentStringsA
	00d0	GetEnvironmentStrings
	008c	FreeEnvironmentStringsW
	00d2	GetEnvironmentStringsW
20	0242	WideCharToMultiByte
	024f	WriteFile
	0164	InitializeCriticalSection
	004f	EnterCriticalSection
	0177	LeaveCriticalSection
25	0153	HeapAlloc
	0159	HeapFree
	0178	LoadLibraryA
	0083	FlushFileBuffers
	0016	CloseHandle
30	0206	SetStdHandle
	01f8	SetFilePointer

### Section Table

35	Section name:	.text
	Virtual Size:	0000379b
	Virtual Address:	00001000
	Size of raw data:	00003800
40	Pointer to Raw Data:	00000400
	Pointer to Relocations:	00000000
	Pointer to Line Numbers:	00000000
	Number of Relocations:	0000
	Number of Line Numbers:	0000
45	Characteristics:	Section contains code Section is executable

Section is readable

5	Section name:	.rdata
	Virtual Size:	000000d3
	Virtual Address:	00005000
	Size of raw data:	00000200
	Pointer to Raw Data:	00003c00
10	Pointer to Relocations:	00000000
	Pointer to Line Numbers:	00000000
	Number of Relocations:	0000
	Number of Line Numbers:	0000
	Characteristics:	Section contains initialized data
15		Section is readable
	Section name:	.data
20	Virtual Size:	00010648
	Virtual Address:	00006000
	Size of raw data:	00004400
	Pointer to Raw Data:	00003e00
	Pointer to Relocations:	00000000
25	Pointer to Line Numbers:	00000000
	Number of Relocations:	0000
	Number of Line Numbers:	0000
	Characteristics:	Section contains initialized data
		Section is readable
30		Section is writeable
	Section name:	.idata
35	Virtual Size:	00000420
	Virtual Address:	00017000
	Size of raw data:	00000600
	Pointer to Raw Data:	00008200
	Pointer to Relocations:	00000000
40	Pointer to Line Numbers:	00000000
	Number of Relocations:	0000
	Number of Line Numbers:	0000
	Characteristics:	Section contains initialized data
		Section is readable
45		Section is writeable

5           Section name: .reloc  
             Virtual Size: 00000ed6  
             Virtual Address: 00018000  
             Size of raw data: 00001000  
             Pointer to Raw Data: 00008800  
             Pointer to Relocations: 00000000  
             Pointer to Line Numbers: 00000000  
 10           Number of Relocations: 0000  
             Number of Line Numbers: 0000  
             Characteristics: Section contains initialized data  
                               Section can be discarded  
                               Section is readable

15

### Header Information

20           Signature: 5a4d  
             Last Page Size: 0090  
             Total Pages in File: 0003  
             Relocation Items: 0000  
             Paragraphs in Header: 0004  
             Minimum Extra Paragraphs: 0000  
 25           Maximum Extra Paragraphs: ffff  
             Initial Stack Segment: 0000  
             Initial Stack Pointer: 00b8  
             Complemented Checksum: 0000  
             Initial Instruction Pointer: 0000  
 30           Initial Code Segment: 0000  
             Relocation Table Offset: 0040  
             Overlay Number: 0000  
             Reserved: 0000 0000 0000 0000  
                       0000 0000 0000 0000  
 35           0000 0000 0000 0000  
             0000 0000 0000 0000  
             Offset to New Header: 00000080  
             Memory Needed: 2K

40

File: Line21.exp

```
L__mTM9R__.edatasdØ
5 @@.drectve_<_
__mTM9F__(4@Q^h__Line21.dllLine21DecodeShiftLeftShiftRight__$(_4_,_8_0_<_
/DLL
/OUT:Line21.dll.edata__&_E_7_ÜLine21DecodeShiftRightShiftLeft_Line21Decode@12_S
hiftright@8_ShiftLeft@8_imp_Line21Decode@12_imp_ShiftLeft@8_imp_ShiftRig
10 ht@8_IMPORT_DESCRIPTOR_Line21__NULL_IMPORT_DESCRIPTOR_Line21_NU
LL_THUNK_DATA
```

Line21.exp

File: Line21.lib

```

!<arch>
5 / 966356238 0 222 '
    _p_~
    _R_p_p_~
    _s_Line21Decode@12_ShiftLeft@8_ShiftRight@8_IMPORT_DESCRIPTOR_Line21_N
    ULL_IMPORT_DESCRIPTOR_imp_Line21Decode@12_imp_ShiftLeft@8_imp_Sh
10 iftRight@8_Line21_NULL_THUNK_DATA/ 966356238 0 244 '
    R_p_s_p_
    ~
    ~
    Line21Decode@12_ShiftLeft@8_ShiftRight@8_IMPORT_DESCRIPTOR_Line21_NULL_IMPORT_DESCRIPTOR_imp_Line21Decode@12_imp_ShiftLeft@8_imp_ShiftRight@8_Line21_NULL_THUNK_DATALine21.dll/ 966356238
15 0 592
    L_mTM9_à_.idata$2_D_X_@_À.idata$6v_X_@
    À_Line21.dll_.idata$2@_À_h.idata$6_.idata$4@_À_h.idata$5@_À_h_8_P_IMPORT_DESCRIPTOR_Line21_NULL_IMPORT_DESCRIPTOR_Line21_NULL_THUNK_DATALine21.dll/ 966356238 0 127
20 L_mTM9P_.idata$3_<@_À_NULL_IMPORT_DESCRIPTOR
    Line21.dll/ 966356238 0 154
    L_mTM9I_.idata$5_d@0À.idata$4_h@0À_Line21_NULL_THUNK_DATALine21.dll/ 966356238 0 492
25 L_mTM9ð_.text_°_°.idata$5_ÄÈ_@_0À.idata$4_ÖÖ_@_0À.idata$6_à@_
    Äÿ%_Line21Decode.text_□`Ä
    ~.idata$5_□`Ä_.idata$4_□`Ä.idata$6_□`Ä_6_imp_Line21Decode
    @12_IMPORT_DESCRIPTOR_Line21Line21.dll/ 966356238 0 484
30 L_mTM9i_.text_°_°.idata$5_ÄÈ_@_0À.idata$4_ÖÖ_@_0À.idata$6_à@_
    Äÿ%_ShiftLeft.text_□`Ä
    ~.idata$5_□`Ä_.idata$4_□`Ä.idata$6_□`Ä_2_imp_ShiftLeft@8_IMPORT_DESCRIPTOR_Line21Line21.dll/ 966356238 0 487
35 L_mTM9î_.text_°_°.idata$5_ÄÈ_@_0À.idata$4_ÖÖ_@_0À.idata$6_à@_
    Äÿ%_ShiftRight.text_□`Ä
    ~.idata$5_□`Ä_.idata$4_□`Ä.idata$6_□`Ä_3_imp_ShiftRight@8_IMPORT_DESCRIPTOR_Line21
40

```

VCPC0...z&amp;□Xü

[illegible]



5  
10  
15  
20  
25  
30  
35  
40  
45



```

5  yyyyyyyyyyyyyyyyyyyJk0__yÿ
   _p_€ñ
   __€_ñ
10  _p_ñ
   ____ñ
   ____ñ__òñ
   _____
   ____
15  ____ñ
   ____ñ
   _p_ñ
   _p_ñ
   _p_€_ñ
20  _p_ñ"_€_MIDL_STUB_MESSAGEñ__
   _____
   _____
   _f_____pfnTranslateToXmitñ____pfnTranslateFromXmitóòñ_____pfnFreeXmit_____
   _____pfnFreeInst&_____XMIT_ROUTINE_QUINTUPLE_____
25  _____
   _____
   ______òñ
   _____
   _F_____pObjóòñ_u___Sizeóòñ_____pfnBind_____
30  _____pfnUnbindòñ&_____GENERIC_BINDING_INFOñ
   _____ñŽ_____
   _____reserved1òñ_"_dataRepresentationñ_____Bufferñ_"_cbBufferóòñ_"_iMe
thod_____reserved2òñ_"_(rpcFlagsóòñ"_____,tagRPCOLEMESSAGEóòñ____u
35  !_____
   #_&"_____pfnAllocate__$_____pfnFree"_____%____MALLOC_FREE_STRUCT&_____
pfnBind_____
   pfnUnbindòñ._'_____GENERIC_BINDING_ROUTINE_PAIRòñ*_____
40  CommOffsetñ_____FaultOffset")_l0_+k™9_____
   _Debug/XDSROM.objF__<Microsoft (R) 32-bit C/C++ Optimizing Compiler
Version 10.00_____MaxLength____~____MinLength_____XDSMes____,_
   _XDSDisp_____:_____
   mpaaRatings____8:_____
45  tpgRatings_X:_____flagmask____`:_____types____à;_____

```

\_\_CType\_\_ XMIT\_ROUTINE\_QUINTUPLE\_\_ GENERIC\_BINDING\_INFO\_\_  
tagRPCOLEMESSAGE\_ &\_ MALLOC\_FREE\_STRUCT" ( \_\_ GENERIC\_BINDING  
ROUTINE\_PAIR\_ &\_ MALLOC\_FREE\_STRUCT\_ \* \_ COMM\_FAULT\_OFFSETS"  
5 \_ FULL\_PTR\_TO\_REFID\_ELEMENT" ( \_\_ GENERIC\_BINDING\_ROUTINE\_PAIR\_\_  
\_ XMIT\_ROUTINE\_QUINTUPLE"\_, \_ PFULL\_PTR\_TO\_REFID\_ELEMENT  
\_1\_ GUID\_3\_ LPCLSID\_6\_ tagENHMETARECORD  
\_1\_ GUID\_L\_ MIDL\_STUB\_DESC\_S\_ FULL\_PTR\_XLAT\_TABLES\_\_ PROP  
SHEETPAGEA\_ f\_ PROPSHEETPAGEW  
\_1\_ CLSID\_, \_ IRpcChannelBufferVtbl\_, \_ DLGTEMPLATE\_ ‡\_ RPC\_SYNTAX\_ID  
10 ENTIFIER\_ @\_ EXCEPTION\_RECORD\_ Ž\_ OVERLAPPED\_ ' \_ FLOATING\_SA  
VE\_AREA  
\_1\_ IID\_, \_ DLGTEMPLATE\_\_ PROPS\_(,žž\$,žž\$,žž\$ \_ HEETPAGEA\_ ' \_ LPDL  
GTEMPLATEA\_ f\_ PROPSHEETPAGEW\_ " \_ tagPOINT\_ —  
\_ CONTEXT\_ ™\_ RPC\_VERSION  
15 \_" \_ POINT\_ — \_ CONTEXT  
\_3\_ LPID\_ %o\_ PEXCEPTION\_RECORD\_ > \_ val\_context  
\_ HDWP  
\_ u\_ PUINT\_\_ RPC\_SS\_THREAD\_HANDLE\_ X\_ CALINFO\_ENUMPROCW\_\_ R  
PC\_AUTH\_IDENTITY\_HANDLE\_ 6\_  
20 ENHMETARECORD&  
\_ SECURITY\_CONTEXT\_TRACKING\_MODE\_ X\_ DATEFMT\_ENUMPROCW\_ t\_  
PSID\_NAME\_USE\_\_ MIDICALLBACK@20" !\_ SECURITY\_DESCRIPTOR\_CONTROL  
— \_ EXPR\_EVAL\_ ž\_ LPFRHOOKPROC\_ X\_ CALINFO\_ENUMPROCA\_ i\_ PFNC  
25 ALLBACK\_ X\_ DATEFMT\_ENUMPROCA\_ □\_ ICMENUMPROCW\_ ©\_  
PQUERYHANDLER  
\_ @\_ FLOAT\_ " \_ LPDWORD\_\_ RPC\_AUTHZ\_HANDLE\_ u\_ MMVERSION  
\_ HRSRC\_\_ RPC\_FORWARD\_FUNCTION@20" \_ @\_ \_ PTOP\_LEVEL\_EXCEPTION\_F  
ILTER  
30 \_ HCONV  
— \_ BYTE\_\_ LONGLONG\_ a\_ LPPROPSHEETPAGEW\_ ±\_ ICMENUMPROCA\_ !\_ LP  
WSTR\_ " \_ ACCESS\_MASK\_ " \_ COLORREF\_ Z\_ LPPROPSHEETPAGEA\_ ' \_ PIMAG  
E\_ TLS\_CALLBACK\_\_ HWAVEOUT\_ \$ \_ PRPC\_RUNDOWN\_\_ HENHMETAFILE  
35 \_ t\_ PBOOL\_ ‡\_ RPC\_SYNTAX\_IDENTIFIER\_ X\_ DESKTOPENUMPROCW\_ " \_ u\_ lo  
ng  
\_ t\_ SPTR\_ !\_ LPOLESTR\_\_ NDR\_CCONTEXT  
\_ u\_ UINT\_ X\_ WNDENUMPROC\_ X\_ DESKTOPENUMPROCA\_ ¶\_ WNDPROC\_ "  
HREFTYPE\_ # \_ DWORDLONG  
40 \_ p\_ LPSTR\_ t  
TOKEN\_TYPE  
\_ !\_ BSTR\_  
RPC\_NS\_HANDLE  
\_ HDESK\_ " \_ PDWORD\_ °\_ EXCEPTION\_POINTERS\_ !  
45 CLIPFORMAT\_ !\_ LPWORD\_ t\_ TYPEKIND\_ X\_ CODEPAGE\_ENUMPROCW

p HPSTR d LPFNPSPCALLBACKW ! OLECHAR X ENUMRESLANGPROC  
t VARKIND X CODEPAGE\_ENUMPROCA ] LPFNPSPCALLBACKA \_\_  
RPC\_BUFPTR X WINSTAENUMPROCW ½ RPC\_AUTH\_KEY\_RETRIEVAL\_FN  
! LPCOLESTR \_\_ SC\_HANDLE  
5 HTASK À REGISTERWORDENUMPROCW X TIMEFMT\_ENUMPROCW \_\_  
X WINSTAENUMPROCA  
! ATOM ! LPCWSTR X DLGPROC À REGISTERWORDENUMPROCA \_\_  
HOLEMENU  
t BOOL \_\_ RPC\_OBJECT\_INQ\_FN@12  
10 " ULONG X TIMEFMT\_ENUMPROCA « PEXCEPTION\_POINTERS Æ tag  
HELPINFO t CALLCONV \_\_ LPMENUTEMPLATEA \_\_ HGLOBAL \_\_ LPVOID  
\_\_  
RPC\_IF\_HANDLE X HOOKPROC  
p PSTR ž LPCFHOOKPROC ! USHORT 7 \_\_  
15 LPHMIDIOUT \_\_ HRESULT \_\_  
PUCHAR \_\_ DISPID È PFULL\_PTR\_XLAT\_TABLES ž LPCCHOOKPROC \_\_  
LPBYTE  
X PROC  
A DATE Đ \_\_  
20 ENHMFENUMPROC \_\_  
PSID  
t LPINT  
Ñ SNB  
p CHAR p LPCSTR Ó SERVER\_ROUTINE  
25 \_\_ SCODE& Ø LPOVERLAPPED\_COMPLETION\_ROUTINE Û \_\_  
YIELDPROC ! VARTYPE \_\_ HWAVEIN ß IRpcChannelBuffer ß IRpc  
ChannelBuffer ž LPPAGEPAINTHOOK t \_\_  
OLERENDER u LPUINT X GOBJENUMPROC \_\_  
HMETAFILEPICT Ñ LPBSTR \_\_  
30 \_\_ byte @ PFLOAT " PSECURITY\_INFORMATION \$ NDR\_RUNDOWN \_\_  
LPCVOID " SERVICE\_STATUS\_HANDLE ž LPPAGESETUPHOOK \_\_  
u u int ! wint t â \_\_  
LPMMIOPROC \_\_  
35 STUB\_THUNK \_\_  
! WCHAR \_\_ HGDIOBJ X ENUMRESNAMEPROC \_\_ HCONVLIST \_\_  
HSZ \_\_  
t INT \_\_ HDDEDATA â tagHANDLETABLE X FARPROC X \_\_  
PROPENUMPROCW " MCIERROR t LPBOOL \_\_  
40 PLONG u WPARAM \_\_  
HMMIO ž \_\_  
LPOFNHOOKPROC ç HPROPSHEETPAGE QUERYHANDLER X \_\_  
PROPENUMPROCA \_\_  
HKL \_\_ HACCEL " \_\_  
45 RPCOLEDATAREP \_\_ VARIANT\_BOOL \_\_ HCURSOR 7 PHANDLE é \_\_  
\_onexit t X \_\_

```

FONTENUMPROCW__" _LCTYPE_î _RPC_DISPATCH_FUNCTION
" _CALID__
RPC_STATUS__ HMIDIIN
    _HDC_ó _MSGBOXCALLBACK_X_
5 FONTENUMPROCA__ _GENERIC_UNBIND_ROUTINE__ _HBITMAP__ _LPA
M__
MENUTEMPLATEW_7_ _LPHMIDISTRM_t _TOKEN_INFORMATION_CLASS__D
RVCALLBACK@20_÷_ _RPC_MESSAGE__
MENUTEMPLATEA
10    _HMIDI_ú _PFNPROPSHEETCALLBACK__ _PSECURITY_DESCRIPTOR
    _HMENU__ _HWINSTA__
    _LCSGAMUTMATCH__ _RPC_CLIENT_FREE@4_ý_ _PMIDL_STUB_DESC
    _LONG__ _NDR_SCONTEXT__
    _ARRAY_INFO_p_va_list_7_ _LPHMIXER_X_    _TIMERPROC_X_
15 DRAWSTATEPROC__ _HINSTANCE_ " _FOURCC__
    _MIDL_STUB_MESSAGE_ " _SECURITY_INFORMATION_X_ _ENUMRESTYPEP
    _ROC_â _HANDLETABLE_7_ _LPHWAVEIN__
    _PFORMAT_STRING_ " _CALTYPE_7_ _LPHANDLE__ _RPC_MGMT_AUTHORI
    _ZATION_FN_! _LANGID" _t _SECURITY_IMPERSONATION_LEVEL_u _SOCKET__
20 _BOOLEAN_7_ _RPC_EP_INQ_HANDLE
    _HWND__ _MMIOPROC@16__ _LPHANDLER_FUNCTION_t _EXCEPTION_DIS
    _POSITION_t _XLAT_SIDE__
    _HHOOK__ _LPTHREAD_START_ROUTINE_7_
    _LPHWAVEOUT__ _HPALETTE_X_ _PROPENUMPROCEXW__ _HMODULE__W
25 AVECALLBACK@20__ _RPC_CLIENT_ALLOC@4__ _HANDLE__ _HMIXER
    _! _PWORD_X_ _PROPENUMPROCEXA__ _HCOLORSPACE_X_ _EDITWORDBR
    _EAKPROCA__ _GENERIC_BINDING_ROUTINE_! _u_short_7_
    _LPHMIDIIN_X_ _LOCALE_ENUMPROCW
    _" _DWORD_X_ _GRAYSTRINGPROC_t _DESKIND
30 _7_ _PHKEY"® _LPTOP_LEVEL_EXCEPTION_FILTER
    _" _LCID__ _LPFNADDPROPSHEETPAGE__ _u_char_X_ _LOCALE_ENUMPROCA
    ___ _HRGN_t _ACL_INFORMATION_CLASS_t _SYSKIND__ _TIMECALLBACK@20_
    _PACCESS_TOKEN__ _SC_LOCK_X_ _LINEDDAPROC__ _PHANDLER_ROUTIN
35 E"! _PSECURITY_DESCRIPTOR_CONTROL_t _FUNCKIND__
    _FXPT2DOT30__ _LPLONG
    _HGLRC__ _HMIDIOUT_t
    _INVOKEKIND__ _HMIDISTRM
    _HFONT__
40 _HPEN__
    _HICON__
    _PVOID__
    _tagMETARECORD_ " _REGSAM_t    _OLEUPDATE_u _MCIDEVICEID__
    _LPSERVICE_MAIN_FUNCTIONW
45 _! _WORD__    _HMETAFILE_X_    _ABORTPROC

```

```

HDRV_ $ _LPSERVICE_MAIN_FUNCTIONA _ž _LPPRINTHOOKPROC _u _si
ze_t _ LCSCSTYPE
_PBYTE _LRESULT_)
MFENUMPROC
5 _é_ FPTR
_HKEY
_UCHAR _PTHREAD_START_ROUTINE _!_ wctype_t _I_RPC_HANDLE
_HDROP _MEMBERID
_SHORT _!_ wchar_t _u _MMRESULT _XMIT_HELPER_ROUTINE _RPC_BI
10 NDING_HANDLE
"_ _HIMC _I_RPC_MUTEX
_HMIXEROBJ _ž _LPSETUPHOOKPROC _RPC_BLOCKING_FUNCTION_
_HBRUSH _X_
SENDASYNCPROC _!_ FILEOP_FLAGS _/_ LPTIMECALLBACK _handle_t _H
15 LOCAL
_t _HFILE _Debug/funct.objF_ <Microsoft (R) 32-bit C/C++ Optimizing Compiler
Version 10.00. _Ø* % _3 _XDSBegin _field _ch _p _mes _1_
cStatePtr _P > 9 _Ë 3 _XDStart _field _ch _p _mes _1_
cStatePtr _Ë > 9 _3 _XDSCont _field _ch _p _mes _1_
20 cStatePtr _@ _F 3 _XDSDData _field _ch _p _mes _1_
cStatePtr *_ ' _E @ _Z 3 _XDSEnd _field _ch _p _mes _1_
cStatePtr _2 _0 $ _ÿ 3 _ControlSave _field _ch _p _mes _1_
cStatePtr _ÿ ½ _Ä 3 _DataSave _field _ch _p _mes _1_
cStatePtr
25 _öÿÿÿt_m
_öÿÿÿt_f
_üÿÿÿt_c _2 _T € {€ _3 _PopOnDecode _field _ch _p _mes _1_ cStatePtr
_üÿÿÿt_f _2 _è _Ö _Đ _3 _RollUpDecode _field _ch _p _mes _1_ cStatePtr
_öÿÿÿt_f
30 _üÿÿÿt_c _2 _p _€ {Ö _3_
PaintOnDecode _field _ch _p _mes _1_ cStatePtr
_üÿÿÿt_f _6 _ü _€ {U _3 _EndOfCaptionDecode _field _ch _p _mes _1_
cStatePtr
_üÿÿÿt_f _€ ^ _YÖ _3_
35 TextDecode _field _ch _p _mes _1_ cStatePtr
_üÿÿÿt_f _ø _)$ _3 _URLDecode _field _ch _p _mes _1_
cStatePtr _2 _€ _l _g\ _3_
ChannelChange _field _ch _p _mes _1_ cStatePtr
_üÿÿÿt_f _Ä _ÿ _Ë 3 _DumpText _field _ch _p _mes _1_ cStatePtr
40 _öÿÿÿt_m
_öÿÿÿt_f
_üÿÿÿt_c _MaxLength _~ _MinLength _XDSMes _
_XDSDisp _à; 4 _CType _XMIT_ROUTINE_QUINTUPLE _GENE
RIC_BINDING_INFO _tagRPCOLEMESSAGE _& _MALLOC_FREE_STRUCT"_(
45 _GENERIC_BINDING_ROUTINE_PAIR _& _MALLOC_FREE_STRUCT *_ _CO
_MM_FAULT_OFFSETS" _FULL_PTR_TO_REFID_ELEMENT" _(_GENERIC_BIND

```

ING\_ROUTINE\_PAIR\_\_\_\_\_XMIT\_ROUTINE\_QUINTUPLE"\_,\_PFULL\_PTR\_TO\_REFI  
D\_ELEMENT  
1\_\_GUID\_3\_\_LPCLSID\_6\_\_tagENHMETARECORD  
1\_\_GUID\_L\_\_MIDL\_STUB\_DESC\_S\_\_FULL\_PTR\_XLAT\_TABLES\_\_\_\_\_PROP  
5 SHEETPAGEA\_f\_\_PROPSHEETPAGEW  
1\_\_CLSID\_7\_\_XDSSState\_,\_IRpcChannelBufferVtbl\_,\_DLGTEMPLATE\_‡\_\_RP  
C\_SYNTAX\_IDENTIFIER\_‡\_\_EXCEPTION\_RECORD\_‡\_\_OVERLAPPED\_'\_\_\_\_  
FLOATING\_SAVE\_AREA  
1\_\_IID\_,\_DLGTEMPLATE\_\_\_\_\_PROPSHEETPAGEA\_'\_\_LPDLGTEMPLATEA\_f  
10 \_\_PROPSHEETPAGEW  
9\_sval\_"\_\_tagPOINT\_\_\_\_CONTEXT\_\_™\_\_RPC\_VERSION  
"\_\_POINT\_\_\_\_CONTEXT  
3\_\_LPIID\_%o\_\_PEXCEPTION\_RECORD  
9\_\_SVAL\_>\_\_val\_context  
15 \_\_HDWP\_H\_cState  
u\_\_PUINT\_\_\_\_\_RPC\_SS\_THREAD\_HANDLE\_X\_\_CALINFO\_ENUMPROCW\_\_\_\_\_R  
PC\_AUTH\_IDENTITY\_HANDLE\_6\_  
ENHMETARECORD&  
\_\_SECURITY\_CONTEXT\_TRACKING\_MODE\_X\_\_DATEFMT\_ENUMPROCW\_t\_  
20 PSID\_NAME\_USE\_\_MIDICALLBACK@20"!\_SECURITY\_DESCRIPTOR\_CONTROL  
\_\_\_\_\_  
EXPR\_EVAL\_‡\_\_LPFRHOOKPROC\_X\_\_CALINFO\_ENUMPROCA\_i\_\_PFNC  
ALLBACK\_X\_\_DATEFMT\_ENUMPROCA\_‡\_\_ICMENUMPROCW\_©\_  
PQUERYHANDLER  
25 \_\_@\_FLOAT\_"\_\_LPDWORD\_\_\_\_\_RPC\_AUTHZ\_HANDLE\_u\_\_MMVERSION  
\_\_\_\_HRSRC\_\_RPC\_FORWARD\_FUNCTION@20"@\_\_PTOP\_LEVEL\_EXCEPTION\_F  
ILTER  
\_\_\_\_HCONV  
\_\_\_\_\_  
30 \_\_BYTE\_\_\_\_LONGLONG\_a\_\_LPPROPSHEETPAGEW\_±\_\_ICMENUMPROCA\_!\_\_LP  
WSTR\_"\_\_ACCESS\_MASK\_"\_\_COLORREF\_Z\_\_LPPROPSHEETPAGEA\_'\_\_PIMAG  
E\_TLS\_CALLBACK\_\_\_\_\_HWAVEOUT\_\$\_\_PRPC\_RUNDOWN\_\_\_\_\_HENHMETAFILE  
\_t\_\_PBOOL\_‡\_\_RPC\_SYNTAX\_IDENTIFIER\_X\_\_DESKTOPENUMPROCW\_"\_\_u\_lo  
ng  
35 \_\_t\_\_SPTR\_!\_\_LPOLESTR\_\_\_\_NDR\_CCONTEXT  
\_\_u\_\_UINT\_X\_\_WNDENUMPROC\_X\_\_DESKTOPENUMPROCA\_¶\_\_WNDPROC\_"\_\_  
HREFTYPE\_#\_\_\_\_DWORDLONG  
\_\_p\_\_LPSTR\_t  
TOKEN\_TYPE  
40 \_\_!\_\_BSTR\_\_\_\_  
RPC\_NS\_HANDLE  
\_\_\_\_HDESK\_"\_\_PDWORD\_°\_\_EXCEPTION\_POINTERS\_!  
CLIPFORMAT\_!\_\_LPWORD\_t\_\_TYPEKIND\_X\_\_CODEPAGE\_ENUMPROCW  
\_\_p\_\_HPSTR\_d\_\_LPFNPPCALLBACKW\_!\_\_OLECHAR\_X\_\_ENUMRESLANGPROC  
45 \_\_t\_\_VARKIND\_X\_\_CODEPAGE\_ENUMPROCA\_]\_\_LPFNPPCALLBACKA\_\_\_\_



```

RPC_BUFPTR_X_WINSTAENUMPROCW_½_RPC_AUTH_KEY_RETRIEVAL_FN
!_LPCOLESTR_____SC_HANDLE
_____HTASK_À_REGISTERWORDENUMPROCW_X_TIMEFMT_ENUMPROCW_
X_WINSTAENUMPROCA
5  !_ATOM!_LPCWSTR_X_DLGPROC_Ã_REGISTERWORDENUMPROCA_____
   _HOLEMENU
   _t_BOOL_RPC_OBJECT_INQ_FN@12
   "_ULONG_X_TIMEFMT_ENUMPROCA_«_PEXCEPTION_POINTERS_Æ_tag
HELPINFO_t_CALLCONV_____LPMENUTEMPLATEA_____HGLOBAL_____LPVOI
10 D_____
   RPC_IF_HANDLE_X_HOOKPROC
   _p_PSTR_ž_LPCFHOOKPROC!_USHORT_7_
   LPHMIDIOUT_____HRESULT_____
   _PUCHAR_____DISPID_È_PFULL_PTR_XLAT_TABLES_ž_LPCCHOOKPROC_____
15 _LPBYTE
   _X_PROC
   _A_DATE_Ð_____
   ENHMFENUMPROC
   _____PSID
20 _t_LPINT
   _Ñ_SNB
   _p_CHAR_p_LPCSTR_Ó_SERVER_ROUTINE
   _____SCODE&Ø_LPOVERLAPPED_COMPLETION_ROUTINE_Û_____
   YIELDPROC!_VARTYPE_____HWAIVEIN_ß_IRpcChannelBuffer_ß_IRpc
25 ChannelBuffer_ž_LPPAGEPAINTHOOK_t
   OLERENDER_u_LPUINT_X_GOBJENUMPROC_____
   HMETAFILEPICT_Ñ_LPBSTR

   _byte_@_PFLOAT_"_PSECURITY_INFORMATION_$_NDR_RUNDOWN_____
30 LPCVOID_"_SERVICE_STATUS_HANDLE_ž_LPPAGESETUPHOOK
   _u_u_int!_wint_t_â_____
   LPMMIOPROC_____
   STUB_THUNK
   !_WCHAR_____HGDIOBJ_X_ENUMRESNAMEPROC_____HCONVLIST
35 _____HSZ
   _t_INT_____HDDEDATA_â_tagHANDLETABLE_X_FARPROC_X_
   PROPENUMPROCW_"_MCIERROR_t_LPBOOL
   _____PLONG_u_WPARAM
   _____HMMIO_ž_____
40 LPOFNHOOKPROC_ç_HPROPSHEETPAGE_QUERYHANDLER_X_
   PROPENUMPROCA
   _____HKL_____HACCEL_"
   RPCOLEDATAREP_____VARIANT_BOOL_____HCURSOR_7_PHANDLE_é_____
   _onexit_t_X_____
45 FONTENUMPROCW_"_LCTYPE_î_RPC_DISPATCH_FUNCTION
   _"_CALID_____

```

RPC\_STATUS\_\_\_\_HMIDIIN  
\_\_\_\_HDC\_ó\_MSGBOXCALLBACK\_X\_  
FONTENUMPROCA\_\_\_\_GENERIC\_UNBIND\_ROUTINE\_\_\_\_HBITMAP\_\_\_\_LPARA  
M\_\_\_\_  
5 MENUTEMPLATEW\_7\_LPHMIDISTRM\_t\_TOKEN\_INFORMATION\_CLASS\_\_D  
RVCALLBACK@20\_÷\_RPC\_MESSAGE\_\_\_\_  
MENUTEMPLATEA  
\_\_\_\_HMIDI\_ú\_PFNPROPSHEETCALLBACK\_\_\_\_PSECURITY\_DESCRIPTOR  
\_\_\_\_HMENU\_\_\_\_HWINSTA\_\_\_\_  
10 LCSGAMUTMATCH\_\_RPC\_CLIENT\_FREE@4\_ý\_PMIDL\_STUB\_DESC  
\_\_\_\_LONG\_\_\_\_NDR\_SCONTEXT\_\_\_\_  
\_\_\_\_ARRAY\_INFO\_p\_va\_list\_7\_LPHMIXER\_X\_\_\_\_TIMERPROC\_X\_\_\_\_  
DRAWSTATEPROC\_\_\_\_HINSTANCE\_\_\_\_FOURCC\_\_\_\_  
\_\_\_\_MIDL\_STUB\_MESSAGE\_\_\_\_SECURITY\_INFORMATION\_X\_ENUMRESTYPEP  
15 ROC\_â\_HANDLETABLE\_7\_LPHWAVEIN\_\_\_\_  
\_\_\_\_PFORMAT\_STRING\_\_\_\_CALTYPE\_7\_LPHANDLE\_\_\_\_RPC\_MGMT\_AUTHORI  
ZATION\_FN\_!\_LANGID"t\_SECURITY\_IMPERSONATION\_LEVEL\_\_\_\_fpos\_t\_u\_S  
OCKET\_\_\_\_BOOLEAN\_7\_RPC\_EP\_INQ\_HANDLE\_\_\_\_  
\_\_\_\_HWND\_\_\_\_MMIOPROC@16\_\_\_\_LPHANDLER\_FUNCTION\_t\_EXCEPTION\_DIS  
20 POSITION\_t\_XLAT\_SIDE\_\_\_\_  
\_\_\_\_HHOOK\_\_\_\_LPTHREAD\_START\_ROUTINE\_7\_\_\_\_  
LPHWAVEOUT\_\_\_\_HPALETTE\_X\_PROPNUMPROCEXW\_\_\_\_HMODULE\_\_\_\_W  
AVECALLBACK@20\_\_RPC\_CLIENT\_ALLOC@4\_\_\_\_HANDLE\_\_\_\_HMIXER  
\_\_\_\_!\_PWORD\_X\_PROPNUMPROCEXA\_\_\_\_HCOLORSPACE\_X\_EDITWORDBR  
25 EAKPROCA\_\_\_\_GENERIC\_BINDING\_ROUTINE\_!\_u\_short\_7\_\_\_\_  
LPHMIDIIN\_X\_LOCALE\_ENUMPROCW\_\_\_\_  
\_\_\_\_"DWORD\_X\_GRAYSTRINGPROC\_t\_DESKIND\_\_\_\_  
\_\_\_\_7\_PHKEY"@\_LPTOP\_LEVEL\_EXCEPTION\_FILTER\_\_\_\_  
\_\_\_\_"LCID\_\_\_\_LPFNADDPROPSHEETPAGE\_\_\_\_u\_char\_X\_LOCALE\_ENUMPROCA\_\_\_\_  
30 \_\_\_\_HRGN\_t\_ACL\_INFORMATION\_CLASS\_t\_SYSKIND\_\_\_\_TIMECALLBACK@20\_\_\_\_  
\_\_\_\_PACCESS\_TOKEN\_\_\_\_SC\_LOCK\_X\_LINEDDAPROC\_\_\_\_PHANDLER\_ROUTIN  
E"!\_PSECURITY\_DESCRIPTOR\_CONTROL\_t\_FUNCKIND\_\_\_\_  
FXPT2DOT30\_\_\_\_LPLONG\_\_\_\_  
35 \_\_\_\_HGLRC\_\_\_\_HMIDIOUT\_t\_\_\_\_  
INVOKEKIND\_\_\_\_HMIDISTRM\_\_\_\_  
\_\_\_\_HFONT\_\_\_\_  
\_\_\_\_HPEN\_\_\_\_  
\_\_\_\_HICON\_\_\_\_  
40 \_\_\_\_PVOID\_\_\_\_  
tagMETARECORD\_"REGSAM\_t\_\_\_\_OLEUPDATE\_u\_MCIDEVICEID\_\_\_\_  
\_\_\_\_LPSERVICE\_MAIN\_FUNCTIONW\_\_\_\_  
\_\_\_\_!\_WORD\_\_\_\_HMETAFILE\_X\_ABORTPROC\_\_\_\_  
\_\_\_\_HDRVR\_\$\_\_LPSERVICE\_MAIN\_FUNCTIONA\_ž\_\_LPPRINTHOOKPROC\_u\_si  
45 ze\_t\_\_\_\_LCSCSTYPE\_\_\_\_  
\_\_\_\_PBYTE\_\_\_\_LRESULT\_\_\_\_)

```

MFENUMPROC
_é_FPTR
_HKEY
_UCHAR_PTHREAD_START_ROUTINE !_wctype_t_I_RPC_HANDLE
5 _HDROP_MEMBERID
_SHORT !_wchar_t_u_MMRESULT_XMIT_HELPER_ROUTINE_RPC_BI
NDING_HANDLE
_"_HIMC_I_RPC_MUTEX
_HMIXEROBJ_ž_LPSETUPHOOKPROC_,_RPC_BLOCKING_FUNCTION_
10 _HBRUSH_X
SENDASYNCPROC !_FILEOP_FLAGS/_LPTIMECALLBACK_handle_t_H
LOCAL
_t_HFILE_<_<_<_J:\Vb60\Line21\funct.c±!_»_Ä_Ê_Ð_p
_P a □ ½ ù_M ^ Ž_Ÿ_Í_û_7_A_F_L_{_É_!_K_U_Z`_Ž_Ů_i_-
15 _□_Ä_Ñ_Ö_#_S_d_i_□_š_Ÿ_¥_´_¼_Ä_İ_Ö_à_ı_U_q_{_€_%_□_ž_®_¼_İ_Ò_à_
ñ_û_
_5:_A_N\ v_,_ž_–_Æ_Ð_Ö_þ_â_ó_"'_5_F_P_U^_e_s_f_‘_¢_§_μ
_Æ_Ð_Ö_þ_â_ó_$_.3_9_C_M_W\ e_l^~!_«'_1_Ä_Ë_Ñ_Ø_â_ò_!+_5_F_p_
u_†_°_μ_¿_Ð_á_æ_÷_
20 _!_2\_a_r_æ_i_«_¼_İ_Ö_ă_ô_2[_}_‡_""#$%&-
./12345689@ABCDEFOPQSTUWYZ[_abcefhiopqry|}~□□,....Š□Ž□‘‘’‘—
™$□_¡_¢_£_¥_§_“_”_©_ª_«_¬_®_±_µ_¶_“_”_‘_’_½_¾_¼_¼_Ä_Ä_Ç_È_É_Ê_Ë_Ï_Î_Ñ_Ó_Ô_Ů_Ý_Þ_ß_à_á_â_ä_å_æ_ï_í_î_ï_ð_ø_ù_ü_ý_þ_
-
!_#_%_&_(*_+/_1_3_5_6_8_:;_>_@_B_C_E_G_H_L_M_
25 N_P_Q_Debug/Parity.objF_<Microsoft (R) 32-bit C/C++ Optimizing Compiler
Version 10.00_
_I_parity_Debug/Tests.objF_<Microsoft (R) 32-bit C/C++ Optimizing Compiler
Version 10.002_Ü_`_3_DefaultTest_field_ch_p_mes_1_
cStatePtr_.T_.)u_3_NullTest_field_ch_p_mes_1_
30 cStatePtr_.İ_9_4f_3_F1XDSTest_field_ch_p_mes_1_
cStatePtr_2_H_F_AŮ_3_XDSSStartTest_field_ch_p_mes_1_
cStatePtr_2_Ä_İ_D"
_3_XDSContTest_field_ch_p_mes_1_ cStatePtr_.</_*k
_3_
35 XDSEndTest_field_ch_p_mes_1_ cStatePtr_2_._=8š
_3_ControlTest_field_ch_p_mes_1_ cStatePtr_2_@_p_k×
_3_TextModeTest_field_ch_p_mes_1_ cStatePtr
_ü_ÿ_ÿ_ÿ_t_c_.Ä_p_kG_3_
RollUpTest_field_ch_p_mes_1_ cStatePtr
40 _ü_ÿ_ÿ_ÿ_t_c_2_L_b_]_3_PaintOnTest_field_ch_p_mes_1_ cStatePtr
_ü_ÿ_ÿ_ÿ_t_c_.Ð_b_]_3_PopOnTest_field_ch_p_mes_1_ cStatePtr
_ü_ÿ_ÿ_ÿ_t_c_6\_b_]_{_3_EndOfCaptionTest_field_ch_p_mes_1_
cStatePtr
_ü_ÿ_ÿ_ÿ_t_c_.à_N_İ_Ý_3_PACTest_field_ch_p_mes_1_ cStatePtr
45 _ü_ÿ_ÿ_ÿ_t_c_2_,_,^_f+_
_3_CtrlChangeTest_field_ch_p_mes_1_ cStatePtr

```

```

_ôÿÿÿt_f_øÿÿÿt_ctrl
_üÿÿÿt_c_*_b_]³
_3_CRTest_____field____ch_p_mes_1_cStatePtr
_üÿÿÿt_c_.^_\_W_3_URLTest_____field____ch_p_mes_1_cStatePtr
5 _üÿÿÿt_c_.$__V_Q_q_3_
URLEndTest_____field____ch_p_mes_1_cStatePtr
_ôÿÿÿt_l
_øÿÿÿt_f
_üÿÿÿp_t_____XMIT_ROUTINE_QUINTUPLE_____GENERIC_BINDING_INFO
10 _____tagRPCOLEMESSAGE_&_MALLOC_FREE_STRUCT"(_GENERIC_BINDING_ROUTINE_PAIR_&_MALLOC_FREE_STRUCT*_COMM_FAULT_OFFSET
S"_.FULL_PTR_TO_REFID_ELEMENT"(_GENERIC_BINDING_ROUTINE_PAIR
_____XMIT_ROUTINE_QUINTUPLE"_,_PFULL_PTR_TO_REFID_ELEMENT
_1_GUID_3_LPCLSID_6_tagENHMETARECORD
15 _1_GUID_L_MIDL_STUB_DESC_S_FULL_PTR_XLAT_TABLES_____PROP
SHEETPAGEA_f_PROPSHEETPAGEW
_1_CLSID_7_XDSSState_,_IRpcChannelBufferVtbl_,_DLGTEMPLATE_‡_RPC_SYNTAX_IDENTIFIER_œ_EXCEPTION_RECORD_ž_OVERLAPPED_‘_
FLOATING_SAVE_AREA
20 _1_IID_,_DLGTEMPLATE_____PROPSHEETPAGEA_’_LPDLGTEMPLATEA_f_
PROPSHEETPAGEW
_9_sval_”_tagPOINT_____CONTEXT_™_RPC_VERSION
”_POINT_____CONTEXT
_3_LPIID_%o_PEXCEPTION_RECORD
25 _9_SVAL_›_val_context
_____HDWP_H_cState
_u_PUINT_____RPC_SS_THREAD_HANDLE_X_CALINFO_ENUMPROCW_____RPC_AUTH_IDENTITY_HANDLE_6_
ENHMETARECORD&_
30 _SECURITY_CONTEXT_TRACKING_MODE_X_DATEFMT_ENUMPROCW_t_
PSID_NAME_USE_____MIDICALLBACK@20"!_SECURITY_DESCRIPTOR_CONTROL
_____
EXPR_EVAL_ž_LPFRHOOKPROC_X_CALINFO_ENUMPROCA_i_PFN_C
ALLBACK_X_DATEFMT_ENUMPROCA_¤_ICMENUMPROCW_©_
35 PQUERYHANDLER
_@_FLOAT_"_LPDWORD_____RPC_AUTHZ_HANDLE_u_MMVERSION
_____HRSRC_____RPC_FORWARD_FUNCTION@20"_®_PTOP_LEVEL_EXCEPTION_FILTER
_____HCONV
40 _____
_BYTE_____LONGLONG_a_LPPROPSHEETPAGEW_±_ICMENUMPROCA_!_LPWSTR_"_ACCESS_MASK_"_COLORREF_Z_LPPROPSHEETPAGEA_’_PIMAGE_TLS_CALLBACK_____HWAVEOUT_$_PRPC_RUNDOWN_____HENHMETAFILE
_t_PBOOL_‡_RPC_SYNTAX_IDENTIFIER_X_DESKTOPENUMPROCW_"_u_lo
45 ng
t_SPTR_!_LPOLESTR_____NDR_CCONTEXT

```

5   \_u\_UINT\_X\_WNDENUMPROC\_X\_DESKTOPENUMPROCA\_¶\_WNDPROC\_"  
HREFTYPE\_#     DWORDLONG  
\_p\_LPSTR\_t  
TOKEN\_TYPE  
!\_BSTR  
RPC\_NS\_HANDLE  
HDESK\_"\_PDWORD\_°\_EXCEPTION\_POINTERS!  
CLIPFORMAT!\_LPWORD\_t\_TYPEKIND\_X\_CODEPAGE\_ENUMPROCW  
\_p\_HPSTR\_d\_LPFNPSPCALLBACKW!\_OLECHAR\_X\_ENUMRESLANGPROC  
10   \_t\_VARKIND\_X\_CODEPAGE\_ENUMPROCA\_]\_LPFNPSPCALLBACKA\_  
RPC\_BUFPTR\_X\_WINSTAENUMPROCW\_½\_RPC\_AUTH\_KEY\_RETRIEVAL\_FN  
!\_LPCOLESTR\_\_\_\_\_SC\_HANDLE  
HTASK\_À\_REGISTERWORDENUMPROCW\_X\_TIMEFMT\_ENUMPROCW\_  
X\_WINSTAENUMPROCA  
15   !\_ATOM!\_LPCWSTR\_X\_DLGPTR\_À\_REGISTERWORDENUMPROCA\_\_\_\_  
HOLEMENU  
\_t\_BOOL\_RPC\_OBJECT\_INQ\_FN@12  
"\_ULONG\_X\_TIMEFMT\_ENUMPROCA\_«\_PEXCEPTION\_POINTERS\_Æ\_tag  
HELPINFO\_t\_CALLCONV\_\_\_\_\_LPMENUTEMPLATEA\_\_\_\_\_HGLOBAL\_\_\_\_\_LPVOI  
20   D\_  
RPC\_IF\_HANDLE\_X\_HOOKPROC  
\_p\_PSTR\_ž\_LPCFHOOKPROC!\_USHORT\_7\_  
LPHMIDIOUT\_\_\_\_\_HRESULT\_  
PUCHAR\_\_\_\_\_DISPID\_È\_PFULL\_PTR\_XLAT\_TABLES\_ž\_LPCCHOOKPROC\_  
25   LPBYTE  
X\_PROC  
\_A\_DATE\_Đ\_  
ENHMFENUMPROC  
PSID  
30   \_t\_LPINT  
\_N\_SNB  
\_p\_CHAR\_p\_LPCSTR\_Ó\_SERVER\_ROUTINE  
\_SCODE&Ø\_LPOVERLAPPED\_COMPLETION\_ROUTINE\_Û\_  
YIELDPROC!\_VARTYPE\_\_\_\_\_HWAVEIN\_ß\_IRpcChannelBuffer\_ß\_IRpc  
35   ChannelBuffer\_ž\_LPPAGEPAINTHOOK\_t  
OLERENDER\_u\_LPUINT\_X\_GOBJENUMPROC\_\_\_\_  
HMETAFILEPICT\_Ñ\_LPBSTR  
\_byte\_@\_PFLOAT\_"\_PSECURITY\_INFORMATION\_\$\_NDR\_RUNDOWN\_\_\_\_  
40   LPCVOID\_"\_SERVICE\_STATUS\_HANDLE\_ž\_LPPAGESETUPHOOK  
\_u\_u\_int!\_wint\_t\_â\_  
LPMMIOPROC\_\_\_\_  
STUB\_THUNK  
!\_WCHAR\_\_\_\_\_HGDIOBJ\_X\_ENUMRESNAMEPROC\_\_\_\_\_HCONVLIST  
45   HSZ  
\_t\_INT\_\_\_\_\_HDDEDATA\_â\_tagHANDLETABLE\_X\_FARPROC\_X\_

PROPENUMPROCW\_\_ " MCIERROR\_\_ t\_\_ LPBOOL  
\_\_ PLONG\_\_ u\_\_ WPARAM  
\_\_ HMMIO\_\_ ž\_\_  
LPOFNHOOKPROC\_\_ ç\_\_ HPROPSHEETPAGE\_\_ QUERYHANDLER\_\_ X\_\_  
5 PROPENUMPROCA  
\_\_ HKL\_\_ HACCEL\_\_ "  
RPCOLEDATAREP\_\_ VARIANT\_\_ BOOL\_\_ HCURSOR\_\_ 7\_\_ PHANDLE\_\_ é\_\_  
\_\_ onexit\_\_ t\_\_ X\_\_  
FONENUMPROCW\_\_ " LCTYPE\_\_ î\_\_ RPC\_DISPATCH\_FUNCTION  
10 " CALID\_\_  
RPC\_STATUS\_\_ HMIDIIN  
\_\_ HDC\_\_ ó\_\_ MSGBOXCALLBACK\_\_ X\_\_  
FONENUMPROCA\_\_ GENERIC\_UNBIND\_ROUTINE\_\_ HBITMAP\_\_ LPARA  
M\_\_  
15 MENUTEMPLATEW\_\_ 7\_\_ LPHMIDISTRM\_\_ t\_\_ TOKEN\_INFORMATION\_CLASS\_\_ D  
RVCALLBACK@20\_\_ ÷\_\_ RPC\_MESSAGE\_\_  
MENUTEMPLATEA  
\_\_ HMIDI\_\_ ú\_\_ PFNPROPSHEETCALLBACK\_\_ PSECURITY\_DESCRIPTOR  
\_\_ HMENU\_\_ HWINSTA\_\_  
20 LCSGAMUTMATCH\_\_ RPC\_CLIENT\_FREE@4\_\_ ý\_\_ PMIDL\_STUB\_DESC  
\_\_ LONG\_\_ NDR\_SCONTEXT\_\_  
ARRAY\_INFO\_\_ p\_\_ va\_list\_\_ 7\_\_ LPHMIXER\_\_ X\_\_ TIMERPROC\_\_ X\_\_  
DRAWSTATEPROC\_\_ HINSTANCE\_\_ " FOURCC\_\_  
\_\_ MIDL\_STUB\_MESSAGE\_\_ " SECURITY\_INFORMATION\_\_ X\_\_ ENUMRESTYPEP  
25 ROC\_\_ â\_\_ HANDLETABLE\_\_ 7\_\_ LPHWAVEIN\_\_  
\_\_ PFORMAT\_STRING\_\_ " CALTYPE\_\_ 7\_\_ LPHANDLE\_\_ RPC\_MGMT\_AUTHORI  
ZATION\_FN\_\_ !\_\_ LANGID\_\_ t\_\_ SECURITY\_IMPERSONATION\_LEVEL\_\_ fpos\_\_ t\_\_ u\_\_ S  
OCKET\_\_ BOOLEAN\_\_ 7\_\_ RPC\_EP\_INQ\_HANDLE\_\_  
\_\_ HWND\_\_ MMIOPROC@16\_\_ LPHANDLER\_FUNCTION\_\_ t\_\_ EXCEPTION\_DIS  
30 POSITION\_\_ t\_\_ XLAT\_SIDE\_\_  
\_\_ HHOOK\_\_ LPTHREAD\_START\_ROUTINE\_\_ 7\_\_  
LPHWAVEOUT\_\_ HPALETTE\_\_ X\_\_ PROPENUMPROCEXW\_\_ HMODULE\_\_ W  
AVECALLBACK@20\_\_ RPC\_CLIENT\_ALLOC@4\_\_ HANDLE\_\_ HMIXER\_\_  
\_\_ !\_\_ PWORD\_\_ X\_\_ PROPENUMPROCEXA\_\_ HCOLORSPACE\_\_ X\_\_ EDITWORDBR  
35 EAKPROCA\_\_ GENERIC\_BINDING\_ROUTINE\_\_ !\_\_ u\_\_ short\_\_ 7\_\_  
LPHMIDIIN\_\_ X\_\_ LOCALE\_ENUMPROCW\_\_  
\_\_ " DWORD\_\_ X\_\_ GRAYSTRINGPROC\_\_ t\_\_ DESCKIND\_\_  
\_\_ 7\_\_ PHKEY\_\_ ®\_\_ LPTOP\_LEVEL\_EXCEPTION\_FILTER\_\_  
\_\_ " LCID\_\_ LPFNADDPROPSHEETPAGE\_\_ u\_\_ char\_\_ X\_\_ LOCALE\_ENUMPROCA  
40 \_\_ HRGN\_\_ t\_\_ ACL\_INFORMATION\_CLASS\_\_ t\_\_ SYSKIND\_\_ TIMECALLBACK@20\_\_  
\_\_  
PACCESS\_TOKEN\_\_ SC\_LOCK\_\_ X\_\_ LINEDDAPROC\_\_ PHANDLER\_ROUTIN  
E\_\_ !\_\_ PSECURITY\_DESCRIPTOR\_CONTROL\_\_ t\_\_ FUNCKIND\_\_  
FXPT2DOT30\_\_ LPLONG\_\_  
45 \_\_ HGLRC\_\_ HMIDIOUT\_\_ t\_\_  
INVOKEKIND\_\_ HMIDISTRM\_\_

```

____ HFONT
____ HPEN
____ HICON
____ PVOID
5 tagMETARECORD__" REGSAM__t OLEUPDATE__u_MCIDEVICEID__
  LPSERVICE_MAIN_FUNCTIONW
  !_WORD__ HMETAFILE__X ABORTPROC
  __HDRV__$_ LPSERVICE_MAIN_FUNCTIONA__ž LPPRINTHOOKPROC__u_si
ze_t__ LCSCSTYPE
10 __PBYTE__LRESULT__)
MFENUMPROC
  é_FPTR
  HKEY
  __UCHAR__PTHREAD_START_ROUTINE__!_wctype_t__I_RPC_HANDLE
15 __HDROP__MEMBERID
  __SHORT__!_wchar_t__u_MMRESULT__XMIT_HELPER_ROUTINE__RPC_BI
NDING_HANDLE
  __HIMC__I_RPC_MUTEX
  HMIXEROBJ__ž LPSETUPHOOKPROC__, RPC_BLOCKING_FUNCTION__
20 __HBRUSH__X
SENDASYNCPROC__!FILEOP_FLAGS__/_LPTIMECALLBACK__handle_t__H
LOCAL
  _t_HFILE__`Æ__<`  Æ__J:\Vb60\Line21\Tests.c_u`  f  p  u  {
  ^  '  _  ž  £  ©  Á  Ě  Đ  ×  Ü  â
25
  -
  -
  "
30 (
  P
  Z
  -
  f
35 k
  q
  %o
  Ž
40 •
  š
  ¼
  Æ
45 Ě
  Ò

```

×  
à  
í  
5    1\_ ; B\_ G\_ P\_ ] ... i «<sup>2</sup> · À Í ð  
      "/Weov{,, 'ÇÑØÝæó\_  
—  
—  
—  
—  
—  
+  
10   4  
     ;  
     H  
     T  
     ~  
15   ¢  
     §  
     ®  
     3  
     ¼  
20   É  
     ñ  
     ÿ  
      + ? [ \_ e \_ l \_ q \_ z \_ □ \_ ” \_ ¥ \_ — \_ Ñ \_ I \_ Á \_ 9 \_ ± \_ » \_ Â \_  
      !\$%&'(),-  
25   ./0156789:=@ACDFGJMNQSTWZ[ ] ^`adghjkmnqtuvwxyz{~□,,,...‡^Ž□□‘”””•~>œžŸ|¢¥”©  
     «~®~²¶;¹¼½¾¿ÀÃÄ\_\_\_\_\_Debug/XDSMessages.objF<Microsoft (R) 32-bit C/C++  
     Optimizing Compiler Version 10.00\* ¼é äP\_K\_Def\_p\_mes\_1\_  
      cStatePtr\_\*\_P\_"\_9\_K\_Vcp\_p\_mes\_1\_  
      cStatePtr\_ðÿÿÿp\_b2\_ðÿÿÿp\_b1\_øÿÿÿp\_type\_üÿÿÿp\_flags\_\*\_°\_ú  
30   ð[\_K\_Typ\_p\_mes\_1\_      cStatePtr  
      üÿÿÿt\_i\_\_\_\_\_XDSMes\_:4\_mpaaratings\_8:4\_tvpgratings\_X:L\_flag  
      mask`\_:4\_types\_\_\_\_\_XMIT\_ROUTINE\_QUINTUPLE\_\_\_\_\_GENERIC\_BINDING  
      INFO\_\_\_\_\_tagRPCOLEMESSAGE\_&\_MALLOC\_FREE\_STRUCT"(\_GENERIC\_  
35   BINDING\_ROUTINE\_PAIR\_&\_MALLOC\_FREE\_STRUCT\*\_COMM\_FAULT\_O  
      FFSETS".\_FULL\_PTR\_TO\_REFID\_ELEMENT"(\_GENERIC\_BINDING\_ROUTINE  
      PAIR\_XMIT\_ROUTINE\_QUINTUPLE",\_PFULL\_PTR\_TO\_REFID\_ELEMENT  
      1\_GUID\_3\_LPCLSID\_6\_tagENHMETARECORD  
      1\_GUID\_L\_MIDL\_STUB\_DESC\_S\_FULL\_PTR\_XLAT\_TABLES\_\_\_\_\_PROP  
      SHEETPAGEA\_f\_PROPSHEETPAGEW  
40   1\_CLSID\_7\_XDSState\_,\_IRpcChannelBufferVtbl\_,\_DLGTEMPLATE\_‡\_RP  
      C\_SYNTAX\_IDENTIFIER\_œ\_EXCEPTION\_RECORD\_Ž\_OVERLAPPED\_‘\_  
      FLOATING\_SAVE\_AREA  
      1\_IID\_,\_DLGTEMPLATE\_\_\_\_\_PROPSHEETPAGEA\_'\_LPDLGTEMPLATEA\_f  
      PROPSHEETPAGEW  
45   9\_sval\_"\_tagPOINT\_\_\_\_\_CONTEXT\_™\_RPC\_VERSION  
      "\_"POINT\_\_\_\_\_CONTEXT



3\_LPIID\_%%\_PEXCEPTION\_RECORD  
9\_SVAL\_>\_val\_context  
HDWP\_H\_cState  
u\_PUINT\_\_\_\_RPC\_SS\_THREAD\_HANDLE\_X\_CALINFO\_ENUMPROCW\_\_\_\_R  
5\_PC\_AUTH\_IDENTITY\_HANDLE\_6\_  
ENHMETARECORD&  
SECURITY\_CONTEXT\_TRACKING\_MODE\_X\_DATEFMT\_ENUMPROCW\_t  
PSID\_NAME\_USE\_\_\_\_MIDICALLBACK@20"!\_SECURITY\_DESCRIPTOR\_CONTROL  
  
10\_EXPR\_EVAL\_ž\_LPFRHOOKPROC\_X\_CALINFO\_ENUMPROCA\_i\_PFNCA  
ALLBACK\_X\_DATEFMT\_ENUMPROCA\_¤\_ICMENUMPROCW\_©\_  
PQUERYHANDLER  
@\_FLOAT\_"\_LPDWORD\_\_\_\_RPC\_AUTHZ\_HANDLE\_u\_MMVERSION  
HRSRC\_\_\_\_RPC\_FORWARD\_FUNCTION@20"®\_PTOP\_LEVEL\_EXCEPTION\_F  
15\_ILTER  
HCONV  
  
\_BYTE\_\_\_\_LONGLONG\_a\_LPPROPSHEETPAGEW\_±\_ICMENUMPROCA\_!\_LP  
WSTR\_"\_ACCESS\_MASK\_"\_COLORREF\_Z\_LPPROPSHEETPAGEA'\_PIMAG  
20\_E\_TLS\_CALLBACK\_\_\_\_HWAVEOUT\_\$\_PRPC\_RUNDOWN\_\_\_\_HENHMETAFILE  
\_t\_PBOOL\_‡\_RPC\_SYNTAX\_IDENTIFIER\_X\_DESKTOPENUMPROCW\_"\_u\_lo  
ng  
\_t\_SPTR\_!\_LPOLESTR\_\_\_\_NDR\_CCONTEXT  
\_u\_UINT\_X\_WNDENUMPROC\_X\_DESKTOPENUMPROCA\_¶\_WNDPROC\_"\_  
25\_HREFTYPE\_#\_DWORDLONG  
\_p\_LPSTR\_t  
TOKEN\_TYPE  
!\_BSTR\_\_\_\_  
RPC\_NS\_HANDLE  
30\_HDESK\_"\_PDWORD\_°\_EXCEPTION\_POINTERS\_!  
CLIPFORMAT\_!\_LPWORD\_t\_TYPEKIND\_X\_CODEPAGE\_ENUMPROCW  
\_p\_HPSTR\_d\_LPFNPSPCALLBACKW\_!\_OLECHAR\_X\_ENUMRESLANGPROC  
\_t\_VARKIND\_X\_CODEPAGE\_ENUMPROCA\_]\_LPFNPSPCALLBACKA\_\_\_\_  
RPC\_BUFPTR\_X\_WINSTAENUMPROCW\_½\_RPC\_AUTH\_KEY\_RETRIEVAL\_FN  
35\_!\_LPCOLESTR\_\_\_\_SC\_HANDLE  
HTASK\_À\_REGISTERWORDENUMPROCW\_X\_TIMEFMT\_ENUMPROCW\_  
X\_WINSTAENUMPROCA  
!\_ATOM\_!\_LPCWSTR\_X\_DLGPTR\_À\_REGISTERWORDENUMPROCA\_\_\_\_  
HOLEMENU  
40\_t\_BOOL\_\_\_\_RPC\_OBJECT\_INQ\_FN@12  
"\_ULONG\_X\_TIMEFMT\_ENUMPROCA\_«\_PEXCEPTION\_POINTERS\_Æ\_tag  
HELPINFO\_t\_CALLCONV\_\_\_\_LPMENUTEMPLATEA\_\_\_\_HGLOBAL\_\_\_\_LPVOI  
D\_\_\_\_  
RPC\_IF\_HANDLE\_X\_HOOKPROC  
45\_p\_PSTR\_ž\_LPCFHOOKPROC\_!\_USHORT\_7\_

LPHMIDIOUT\_\_\_\_HRESULT\_\_\_\_  
\_\_\_\_PUCHAR\_\_\_\_DISPID\_È\_\_\_\_PFULL\_PTR\_XLAT\_TABLES\_\_\_\_ž\_\_\_\_LPCCHOOKPROC\_\_\_\_  
\_\_\_\_LPBYTE\_\_\_\_  
\_\_\_\_X\_\_\_\_PROC\_\_\_\_  
5 \_\_\_\_A\_\_\_\_DATE\_\_\_\_Đ\_\_\_\_  
ENHMFENUMPROC\_\_\_\_  
\_\_\_\_PSID\_\_\_\_  
\_\_\_\_t\_\_\_\_LPINT\_\_\_\_  
\_\_\_\_Ñ\_\_\_\_SNB\_\_\_\_  
10 \_\_\_\_p\_\_\_\_CHAR\_\_\_\_p\_\_\_\_LPCSTR\_\_\_\_Ó\_\_\_\_SERVER\_ROUTINE\_\_\_\_  
\_\_\_\_SCODE&\_Ø\_\_\_\_LPOVERLAPPED\_COMPLETION\_ROUTINE\_\_\_\_Û\_\_\_\_  
\_\_\_\_YIELDPROC\_\_\_\_!\_\_\_\_VARTYPE\_\_\_\_HWAVERIN\_\_\_\_ß\_\_\_\_IRpcChannelBuffer\_\_\_\_ß\_\_\_\_IRpc  
ChannelBuffer\_\_\_\_ž\_\_\_\_LPPAGEPAINTHOOK\_\_\_\_t\_\_\_\_  
\_\_\_\_OLERENDER\_\_\_\_u\_\_\_\_LPUINT\_\_\_\_X\_\_\_\_GOBJENUMPROC\_\_\_\_  
15 HMETAFILEPICT\_\_\_\_Ñ\_\_\_\_LPBSTR\_\_\_\_  
\_\_\_\_byte\_\_\_\_@\_\_\_\_PFLOAT\_\_\_\_"\_\_\_\_PSECURITY\_INFORMATION\_\_\_\_\$\_\_\_\_NDR\_RUNDOWN\_\_\_\_  
LPCVOID\_\_\_\_"\_\_\_\_SERVICE\_STATUS\_HANDLE\_\_\_\_ž\_\_\_\_LPPAGESETUPHOOK\_\_\_\_  
\_\_\_\_u\_\_\_\_u\_\_\_\_int\_\_\_\_!\_\_\_\_wint\_\_\_\_t\_\_\_\_â\_\_\_\_  
20 LPMMIOPROC\_\_\_\_  
STUB\_THUNK\_\_\_\_  
\_\_\_\_!\_\_\_\_WCHAR\_\_\_\_HGDIOBJ\_\_\_\_X\_\_\_\_ENUMRESNAMEPROC\_\_\_\_HCONVLIST\_\_\_\_  
\_\_\_\_HSZ\_\_\_\_  
\_\_\_\_t\_\_\_\_INT\_\_\_\_HDDEDATA\_\_\_\_â\_\_\_\_tagHANDLETABLE\_\_\_\_X\_\_\_\_FARPROC\_\_\_\_X\_\_\_\_  
25 PROPENUMPROCW\_\_\_\_"\_\_\_\_MCIERROR\_\_\_\_t\_\_\_\_LPBOOL\_\_\_\_  
\_\_\_\_PLONG\_\_\_\_u\_\_\_\_WPARAM\_\_\_\_  
\_\_\_\_HMMIO\_\_\_\_ž\_\_\_\_  
LPOFNHOOKPROC\_\_\_\_ç\_\_\_\_HPROPSHEETPAGE\_\_\_\_QUERYHANDLER\_\_\_\_X\_\_\_\_  
PROPENUMPROCA\_\_\_\_  
30 \_\_\_\_HKL\_\_\_\_HACCEL\_\_\_\_"\_\_\_\_  
RPCOLEDATAREP\_\_\_\_VARIANT\_BOOL\_\_\_\_HCURSOR\_\_\_\_7\_\_\_\_PHANDLE\_\_\_\_é\_\_\_\_  
\_\_\_\_onexit\_\_\_\_t\_\_\_\_X\_\_\_\_  
FONTENUMPROCW\_\_\_\_"\_\_\_\_LCTYPE\_\_\_\_î\_\_\_\_RPC\_DISPATCH\_FUNCTION\_\_\_\_  
\_\_\_\_"\_\_\_\_CALID\_\_\_\_  
35 RPC\_STATUS\_\_\_\_HMIDIIN\_\_\_\_  
\_\_\_\_HDC\_\_\_\_ó\_\_\_\_MSGBOXCALLBACK\_\_\_\_X\_\_\_\_  
FONTENUMPROCA\_\_\_\_GENERIC\_UNBIND\_ROUTINE\_\_\_\_HBITMAP\_\_\_\_LPARAM\_\_\_\_  
MENUTEMPLATEW\_\_\_\_7\_\_\_\_LPHMIDISTRM\_\_\_\_t\_\_\_\_TOKEN\_INFORMATION\_CLASS\_\_\_\_D\_\_\_\_  
40 RVCALLBACK@20\_\_\_\_÷\_\_\_\_RPC\_MESSAGE\_\_\_\_  
MENUTEMPLATEA\_\_\_\_  
\_\_\_\_HMIDI\_\_\_\_ú\_\_\_\_PFNPROPSHEETCALLBACK\_\_\_\_PSECURITY\_DESCRIPTOR\_\_\_\_  
\_\_\_\_HMENU\_\_\_\_HWINSTA\_\_\_\_  
LCSGAMUTMATCH\_\_\_\_RPC\_CLIENT\_FREE@4\_\_\_\_ý\_\_\_\_PMIDL\_STUB\_DESC\_\_\_\_  
45 \_\_\_\_LONG\_\_\_\_NDR\_SCONTEXT\_\_\_\_  
ARRAY\_INFO\_\_\_\_p\_\_\_\_va\_list\_\_\_\_7\_\_\_\_LPHMIXER\_\_\_\_X\_\_\_\_TIMERPROC\_\_\_\_X\_\_\_\_

DRAWSTATEPROC\_\_\_\_\_ HINSTANCE\_ " FOURCC\_  
\_\_\_\_MIDL\_STUB\_MESSAGE\_ " SECURITY\_INFORMATION\_X\_ENUMRESTYPEP  
ROC\_â\_HANDLETABLE\_7\_ LPHWAVEIN\_  
\_\_\_\_PFORMAT\_STRING\_ " CALTYPE\_7\_LPHANDLE\_\_\_\_\_RPC\_MGMT\_AUTHORI  
5 ZATION\_FN\_! LANGID" t SECURITY\_IMPERSONATION\_LEVEL\_\_\_\_\_fpos\_t\_u\_S  
OCKET\_\_\_\_\_BOOLEAN\_7\_RPC\_EP\_INQ\_HANDLE  
\_\_\_\_HWND\_\_\_\_\_MMIOPROC@16\_\_\_\_\_LPHANDLER\_FUNCTION\_t\_EXCEPTION\_DIS  
POSITION\_t\_XLAT\_SIDE  
\_\_\_\_HHOOK\_\_\_\_\_LPTHREAD\_START\_ROUTINE\_7\_  
10 LPHWAVEOUT\_\_\_\_\_HPALETTE\_X\_PROPENUMPROCEXW\_\_\_\_\_HMODULE\_\_\_\_\_W  
AVECALLBACK@20\_\_\_\_\_RPC\_CLIENT\_ALLOC@4\_\_\_\_\_HANDLE\_\_\_\_\_HMIXER  
\_\_\_\_!\_PWORD\_X\_PROPENUMPROCEXA\_\_\_\_\_HCOLORSPACE\_X\_EDITWORDBR  
EAKPROCA\_\_\_\_\_GENERIC\_BINDING\_ROUTINE\_!\_u\_short\_7\_  
LPHMIDIIN\_X\_LOCALE\_ENUMPROCW  
15 " DWORD\_X\_GRAYSTRINGPROC\_t\_DESKIND  
\_\_\_\_7\_PHKEY"®\_LPTOP\_LEVEL\_EXCEPTION\_FILTER  
\_\_\_\_" LCID\_\_\_\_\_LPFNADDPROPSHEETPAGE\_\_\_\_\_u\_char\_X\_LOCALE\_ENUMPROCA  
\_\_\_\_HRGN\_t\_ACL\_INFORMATION\_CLASS\_t\_SYSKIND\_\_\_\_\_TIMECALLBACK@20\_  
\_\_\_\_\_  
20 PACCESS\_TOKEN\_\_\_\_\_SC\_LOCK\_X\_LINEDDAPROC\_\_\_\_\_PHANDLER\_ROUTIN  
E"!\_PSECURITY\_DESCRIPTOR\_CONTROL\_t\_FUNCKIND\_\_\_\_\_  
FXPT2DOT30\_\_\_\_\_LPLONG  
\_\_\_\_HGLRC\_\_\_\_\_HMIDIOUT\_t  
INVOKEKIND\_\_\_\_\_HMIDISTRM  
25 \_\_\_\_\_HFONT  
\_\_\_\_HPEN  
\_\_\_\_HICON  
\_\_\_\_PVOID\_\_\_\_\_  
tagMETARECORD\_ " REGSAM\_t\_\_\_\_\_OLEUPDATE\_u\_MCIDEVICEID\_  
30 \_\_\_\_\_LPSERVICE\_MAIN\_FUNCTIONW  
\_\_\_\_!\_WORD\_\_\_\_\_HMETAFILE\_X\_ABORTPROC  
\_\_\_\_HDRV\_R\_\$ \_LPSERVICE\_MAIN\_FUNCTIONA\_ž \_LPPRINTHOOKPROC\_u\_si  
ze\_t\_\_\_\_\_LCSCSTYPE  
\_\_\_\_PBYTE\_\_\_\_\_LRESULT\_)\_\_\_\_\_  
35 MFENUMPROC  
\_\_\_\_é\_FPTR  
\_\_\_\_HKEY  
\_\_\_\_\_UCHAR\_\_\_\_\_PTHREAD\_START\_ROUTINE\_!\_wctype\_t\_\_\_\_\_I\_RPC\_HANDLE  
\_\_\_\_HDROP\_\_\_\_\_MEMBERID  
40 \_\_\_\_\_SHORT\_!\_wchar\_t\_u\_MMRESULT\_\_\_\_\_XMIT\_HELPER\_ROUTINE\_\_\_\_\_RPC\_BI  
NDING\_HANDLE  
\_\_\_\_" HIMC\_\_\_\_\_I\_RPC\_MUTEX\_\_\_\_\_  
\_\_\_\_HMIXEROBJ\_ž \_LPSETUPHOOKPROC\_\_\_\_,\_\_\_\_RPC\_BLOCKING\_FUNCTION\_\_\_\_\_  
\_\_\_\_HBRUSH\_X\_\_\_\_\_  
45 SENDASYNCPROC\_!\_FILEOP\_FLAGS\_/ \_LPTIMECALLBACK\_\_\_\_\_handle\_t\_\_\_\_\_H  
LOCAL

```

t_HFILE_P_T_DP_T_J:\Vb60\Line21\XDSMessages.c_P_V_g_
"ÿ49B_S_f°Ýé.@_Q1x^~_»ÌØéð#/ @_E_V[_d_u_¥_¶
ø:K_P"#%')*+,-./01234568:<?BCDEGHIJ
Debug\L21Dec.objF_<Microsoft (R) 32-bit C/C++ Optimizing Compiler Version
5 10.00._ä_□_M_DllMain_hInst_"_ul_reason_being_called_
lpReserved_2_d_§_O_Line21Decode_field_ch_p_mes
_øÿÿÿt_j
_üÿÿÿt_i_2_àÊ_ÁÁ_P_stateMachine_field_
_ch_p_mes_üÿÿÿE_cstate_.4_<_R_
10 ShiftRight_"_value_shift_.^_¥_R_ShiftLeft_"_value_shift_.0
_S_DataState_°□_H_CCstate_\l_1_
cStatePtr_XMIT_ROUTINE_QUINTUPLE_GENERIC_BINDING_IN
FO_tagRPCOLEMESSAGE_&_MALLOC_FREE_STRUCT"(_GENERIC_BIN
DING_ROUTINE_PAIR_&_MALLOC_FREE_STRUCT*_COMM_FAULT_OFFS
15 ETS"_.FULL_PTR_TO_REFID_ELEMENT"(_GENERIC_BINDING_ROUTINE_P
AIR_XMIT_ROUTINE_QUINTUPLE",_PFULL_PTR_TO_REFID_ELEMENT
_1_GUID_3_LPCLSID_6_tagENHMETARECORD
_1_GUID_L_MIDL_STUB_DESC_S_FULL_PTR_XLAT_TABLES_PROP
SHEETPAGEA_f_PROPSHEETPAGEW
20 _1_CLSID_7_XDSState_,_IRpcChannelBufferVtbl_,_DLGTEMPLATE_‡_RP
C_SYNTAX_IDENTIFIER_œ_EXCEPTION_RECORD_Ž_OVERLAPPED_‘_
FLOATING_SAVE_AREA
_1_IID_,_DLGTEMPLATE_PROPSHEETPAGEA_,_LPDLGTEMPLATEA_f_
_PROPSHEETPAGEW_"_tagPOINT_—_CONTEXT_™_RPC_VERSION
25 " _POINT_—_CONTEXT
_3_LPIID_%o_PEXCEPTION_RECORD_>_val_context
_HDWP_H_cState
_u_PUINT_RPC_SS_THREAD_HANDLE_X_CALINFO_ENUMPROCW____R
PC_AUTH_IDENTITY_HANDLE_6_
30 ENHMETARECORD&_
_SECURITY_CONTEXT_TRACKING_MODE_X_DATEFMT_ENUMPROCW_t_
PSID_NAME_USE_MIDICALLBACK@20"!_SECURITY_DESCRIPTOR_CONTROL
_____
EXPR_EVAL_ž_LPFRHOOKPROC_X_CALINFO_ENUMPROCA_i_PFN
35 ALLBACK_X_DATEFMT_ENUMPROCA_□_ICMENUMPROCW_©_
PQUERYHANDLER
@_FLOAT_"_LPDWORD_RPC_AUTHZ_HANDLE_u_MMVERSION
_HRSRC_RPC_FORWARD_FUNCTION@20"_®_PTOP_LEVEL_EXCEPTION_F
ILTER
40 ____HCONV
_____
_BYTE____LONGLONG_a_LPPROPSHEETPAGEW_±_ICMENUMPROCA_!_LP
WSTR_"_ACCESS_MASK_"_COLORREF_Z_LPPROPSHEETPAGEA_'_PIMAG
E_TLS_CALLBACK____HWAVEOUT_$_PRPC_RUNDOWN____HENHMETAFI
45 _t_PBOOL_‡_RPC_SYNTAX_IDENTIFIER_X_DESKTOPENUMPROCW_"_u_lo
ng

```

t \_SPTR\_ !\_LPOLESTR\_ \_NDR\_CONTEXT  
\_u\_UINT\_ \_X\_ \_WNDENUMPROC\_ \_X\_ \_DESKTOPENUMPROCA\_ \_!\_ \_WNDPROC\_ \_"  
HREFTYPE\_ # \_DWORDLONG  
\_p\_LPSTR\_ \_t  
5 TOKEN\_TYPE  
\_!\_ \_BSTR\_ \_  
RPC\_NS\_HANDLE  
\_HDESK\_ \_" \_PDWORD\_ \_°\_ \_EXCEPTION\_POINTERS\_ \_!  
CLIPFORMAT\_ !\_LPWORD\_ \_t\_ TYPEKIND\_ \_X\_ \_CODEPAGE\_ENUMPROCW  
10 \_p\_HPSTR\_ \_d\_ \_LPFNPSPCALLBACKW\_ !\_OLECHAR\_ \_X\_ \_ENUMRESLANGPROC  
\_t\_ VARKIND\_ \_X\_ \_CODEPAGE\_ENUMPROCA\_ \_]\_ \_LPFNPSPCALLBACKA\_ \_  
RPC\_BUFPTR\_ \_X\_ \_WINSTAENUMPROCW\_ \_½\_ \_RPC\_AUTH\_KEY\_RETRIEVAL\_FN  
\_!\_ \_LPCOLESTR\_ \_SC\_HANDLE  
\_HTASK\_ \_À\_ \_REGISTERWORDENUMPROCW\_ \_X\_ \_TIMEFMT\_ENUMPROCW\_ \_  
15 X\_ \_WINSTAENUMPROCA\_ \_  
\_!\_ \_ATOM\_ !\_LPCWSTR\_ \_X\_ \_DLGPROC\_ \_Ã\_ \_REGISTERWORDENUMPROCA\_ \_  
\_HOLEMENU\_ \_  
\_t\_ \_BOOL\_ \_RPC\_OBJECT\_INQ\_FN@12  
\_ " \_ULONG\_ \_X\_ \_TIMEFMT\_ENUMPROCA\_ \_«\_ \_PEXCEPTION\_POINTERS\_ \_Æ\_ \_tag  
20 HELPINFO\_ \_t\_ \_CALLCONV\_ \_LPMENUTEMPLATEA\_ \_HGLOBAL\_ \_LPVOID\_ \_  
D\_ \_  
RPC\_IF\_HANDLE\_ \_X\_ \_HOOKPROC\_ \_  
\_p\_ \_PSTR\_ \_ž\_ \_LPCFHOOKPROC\_ !\_USHORT\_ \_7\_ \_  
LPHMIDIOUT\_ \_HRESULT\_ \_  
25 \_PUCHAR\_ \_DISPID\_ \_È\_ \_PFULL\_PTR\_XLAT\_TABLES\_ \_ž\_ \_LPCCHOOKPROC\_ \_  
\_LPBYTE\_ \_  
\_X\_ \_PROC\_ \_  
\_A\_ \_DATE\_ \_Ð\_ \_  
ENHMFENUMPROC  
30 \_PSID\_ \_  
\_t\_ \_LPINT\_ \_  
\_Ñ\_ \_SNB\_ \_  
\_p\_ \_CHAR\_ \_p\_ \_LPCSTR\_ \_Ó\_ \_SERVER\_ROUTINE\_ \_  
\_SCODE&\_ \_Ø\_ \_LPOVERLAPPED\_COMPLETION\_ROUTINE\_ \_Û\_ \_  
35 YIELDPROC\_ !\_VARTYPE\_ \_HWAIVEIN\_ \_ß\_ \_IRpcChannelBuffer\_ \_ß\_ \_IRpc  
ChannelBuffer\_ \_ž\_ \_LPPAGEPAINTHOOK\_ \_t\_ \_  
OLERENDER\_ \_u\_ \_LPUINT\_ \_X\_ \_GOBJENUMPROC\_ \_  
HMETAFILEPICT\_ \_Ñ\_ \_LPBSTR\_ \_  
40 \_byte\_ \_@\_ \_PFLOAT\_ \_" \_PSECURITY\_INFORMATION\_ \_\$ \_NDR\_RUNDOWN\_ \_  
LPCVOID\_ \_" \_SERVICE\_STATUS\_HANDLE\_ \_ž\_ \_LPPAGESETUPHOOK\_ \_  
\_u\_ \_u\_int\_ !\_wint\_ \_t\_ \_â\_ \_  
LPMMIOPROC\_ \_  
STUB\_THUNK  
45 \_9\_ \_sval\_ \_  
\_!\_ \_WCHAR\_ \_HGDI OBJ\_ \_X\_ \_ENUMRESNAMEPROC\_ \_HCONVLIST

```

    HSZ
    _t_INT___ HDDEDATA_&_tagHANDLETABLE_X_FARPROC_X_
    PROPENUMPROCW_"_MCIERROR_t_LPBOOL
    _PLONG_u_WPARAM
5    _HMMIO_ž_
    LPOFNHOOKPROC_ç_HPROPSHEETPAGE_QUERYHANDLER_X_
    PROPENUMPROCA
    _HKL___HACCEL_"
    RPCOLEDATAREP___VARIANT_BOOL___HCURSOR_7_PHANDLE_é_
10    _onexit_t_X_
    FONTENUMPROCW_"_LCTYPE_i_RPC_DISPATCH_FUNCTION
    "_CALID___
    RPC_STATUS___HMIDIIN
    _HDC_ó_MSGBOXCALLBACK_X_
15    FONTENUMPROCA___GENERIC_UNBIND_ROUTINE___HBITMAP___LPARA
    M___
    MENUTEMPLATEW_7_LPHMIDISTRM_t_TOKEN_INFORMATION_CLASS___D
    RVCALLBACK@20÷_RPC_MESSAGE___
    MENUTEMPLATEA
20    _HMIDI_ú_PFNPROPSHEETCALLBACK___PSECURITY_DESCRIPTOR
    _HMENU___HWINSTA___
    LCSGAMUTMATCH___RPC_CLIENT_FREE@4_ý_PMIDL_STUB_DESC
    _LONG___NDR_SCONTEXT___
    ARRAY_INFO_p_va_list_7_LPHMIXER_X___TIMERPROC_X_
25    DRAWSTATEPROC___HINSTANCE_"_FOURCC___
    _MIDL_STUB_MESSAGE_"_SECURITY_INFORMATION_X_ENUMRESTYPEP
    ROC_&_HANDLETABLE_7_LPHWAVEIN___
    _PFORMAT_STRING_"_CALTYPE_7_LPHANDLE___RPC_MGMT_AUTHORI
    ZATION_FN_!_LANGID"t_SECURITY_IMPERSONATION_LEVEL___fpos_t_u_S
30    OCKET___BOOLEAN_7_RPC_EP_INQ_HANDLE
    _HWND___MMIOPROC@16___LPHANDLER_FUNCTION_t_EXCEPTION_DIS
    POSITION_t_XLAT_SIDE
    _HHOOK___LPTHREAD_START_ROUTINE_7_
    LPHWAVEOUT___HPALETTE_X_PROPENUMPROCEXW___HMODULE___W
35    AVECALLBACK@20___RPC_CLIENT_ALLOC@4___HANDLE___HMIXER
    _!_PWORD_X_PROPENUMPROCEXA___HCOLORSPACE_X_EDITWORDBR
    EAKPROCA___GENERIC_BINDING_ROUTINE_!_u_short_7_
    LPHMIDIIN_X_LOCALE_ENUMPROCW
    "_DWORD_X_GRAYSTRINGPROC_t_DESKIND
40    _7_PHKEY"@_LPTOP_LEVEL_EXCEPTION_FILTER
    "_LCID___LPFNADDPROPSHEETPAGE_u_char_X_LOCALE_ENUMPROCA
    ___HRGN_t_ACL_INFORMATION_CLASS_t_SYSKIND___TIMECALLBACK@20_
    _
    PACCESS_TOKEN___SC_LOCK_X_LINEDDAPROC___PHANDLER_ROUTIN
45    E"!_PSECURITY_DESCRIPTOR_CONTROL_t_FUNCKIND___
    FXPT2DOT30___LPLONG

```

```

HGLRC HMIDIOUT_t
INVOKEKIND HMIDISTRM
HFONT
HPEN
5 HICON
PVOID
tagMETARECORD _" REGSAM_t OLEUPDATE_u MCIDEVICEID_
LPSERVICE_MAIN_FUNCTIONW
!_WORD HMETAFILE_X ABORTPROC
10 HDRVR_$ LPSERVICE_MAIN_FUNCTIONA_ž LPPRINTHOOKPROC_u_si
ze_t LCSCSTYPE
PBYTE LRESULT_)
MFENUMPROC
é_FPTR
15 HKEY
_UCHAR_PTHREAD_START_ROUTINE!_wctype_t_I_RPC_HANDLE
HDROP_MEMBERID
_SHORT!_wchar_t_u_MMRESULT_XMIT_HELPER_ROUTINE_RPC_BI
NDING_HANDLE
20 _" HIMC_I_RPC_MUTEX
HMIXEROBJ_ž LPSETUPHOOKPROC_, RPC_BLOCKING_FUNCTION_
HBRUSH_X
SENDASYNCPROC!_FILEOP_FLAGS/_LPTIMECALLBACK_handle_t_H
LOCAL
25 _t_HFILE □ ¾ <□ ¾ J:\Vb60\Line21\L21Dec.c *□ _
_§ ° ª Ä Ö â î ú + G q v □ ™ Ÿ ³ ° Á Ê Ÿ = B_X ] b_x } □ † < ‘
ž ¥ « , !"#%&')+,-/04@CEGHIJLNOPRSVWX[\]
Debug/stable.objF_<Microsoft (R) 32-bit C/C++ Optimizing Compiler Version
10.00_0
30 _T_DataState_~
_U_ControlState_□_V_
XDSSStartState_°_V_XDSContState_D_W_XDSDataState_
_X_XDSEndState_0_Y_URLDataState_XMIT_ROUTINE_QUINTUPLE
GENERIC_BINDING_INFO_tagRPCOLEMESSAGE_&_MALLOC_FREE_STRU
35 CT"(_GENERIC_BINDING_ROUTINE_PAIR_&_MALLOC_FREE_STRUCT_*
_COMM_FAULT_OFFSETS"._FULL_PTR_TO_REFID_ELEMENT"(_GENERIC
_BINDING_ROUTINE_PAIR_XMIT_ROUTINE_QUINTUPLE",_PFULL_PTR_TO
_REFID_ELEMENT
1_GUID_3_LPCLSID_6_tagENHMETARECORD
40 1_GUID_L_MIDL_STUB_DESC_S_FULL_PTR_XLAT_TABLES_PROP
SHEETPAGEA_f_PROPSHEETPAGEW
1_CLSID_,_IRpcChannelBufferVtbl_,_DLGTEMPLATE_‡_RPC_SYNTAX_ID
ENTIFIER_œ_EXCEPTION_RECORD_Ž_OVERLAPPED_‘_FLOATING_SA
VE_AREA
45 1_IID_,_DLGTEMPLATE_PROPSHEETPAGEA_’_LPDLGTEMPLATEA_f
_PROPSHEETPAGEW_"_tagPOINT_—_CONTEXT_™_RPC_VERSION

```

” POINT — CONTEXT  
3 LPIID %\_ PEXCEPTION\_RECORD > \_val\_context  
HDWP  
u PUINT RPC\_SS\_THREAD\_HANDLE\_X CALINFO\_ENUMPROCW \_\_\_ R  
5 PC\_AUTH\_IDENTITY\_HANDLE\_6\_  
ENHMETARECORD&  
SECURITY\_CONTEXT\_TRACKING\_MODE\_X DATEFMT\_ENUMPROCW\_t  
PSID\_NAME\_USE MIDICALLBACK@20"! SECURITY\_DESCRIPTOR\_CONTROL  
10 EXPR\_EVAL\_z LPFRHOOKPROC\_X CALINFO\_ENUMPROCA\_i PFNC  
ALLBACK\_X DATEFMT\_ENUMPROCA\_⊘ ICMENUMPROCW\_©\_  
PQUERYHANDLER  
@\_ FLOAT " LPDWORD RPC\_AUTHZ\_HANDLE\_u MMVERSION  
HRSRC RPC\_FORWARD\_FUNCTION@20" \_®\_ PTOP\_LEVEL\_EXCEPTION\_F  
15 ILTER  
HCONV  
\_ BYTE \_ LONGLONG\_a LPPROPSHEETPAGEW ± ICMENUMPROCA !\_ LP  
WSTR " ACCESS\_MASK " COLORREF\_Z LPPROPSHEETPAGEA ' PIMAG  
20 E\_TLS\_CALLBACK HWAVEOUT \$ PRPC\_RUNDOWN HENHMETAFILE  
\_t\_ PBOOL\_±\_ RPC\_SYNTAX\_IDENTIFIER\_X DESKTOPENUMPROCW " \_u lo  
ng  
\_t\_ SPTR !\_ LPOLESTR \_\_\_ NDR\_CCONTEXT  
\_u\_UINT\_X WNDENUMPROC\_X DESKTOPENUMPROCA\_¶\_ WNDPROC "  
25 HREFTYPE # DWORDLONG  
\_p\_ LPSTR\_t  
TOKEN\_TYPE  
!\_ BSTR  
RPC\_NS\_HANDLE  
30 HDESK " PDWORD ° EXCEPTION\_POINTERS !  
CLIPFORMAT !\_ LPWORD\_t TYPEKIND\_X CODEPAGE\_ENUMPROCW  
\_p\_ HPSTR\_d LPFNPSPCALLBACKW !\_ OLECHAR\_X ENUMRESLANGPROC  
\_t\_ VARKIND\_X CODEPAGE\_ENUMPROCA\_] LPFNPSPCALLBACKA\_  
RPC\_BUFPTR\_X WINSTAENUMPROCW\_½\_ RPC\_AUTH\_KEY\_RETRIEVAL\_FN  
35 !\_ LPCOLESTR \_\_\_ SC\_HANDLE  
HTASK\_Ä REGISTERWORDENUMPROCW\_X TIMEFMT\_ENUMPROCW\_  
X WINSTAENUMPROCA  
!\_ ATOM !\_ LPCWSTR\_X DLGPROC\_Ä REGISTERWORDENUMPROCA\_  
HOLEMENU  
40 \_t\_ BOOL RPC\_OBJECT\_INQ\_FN@12  
" ULONG\_X TIMEFMT\_ENUMPROCA\_«\_ PEXCEPTION\_POINTERS\_Æ\_tag  
HELPINFO\_t CALLCONV LPMENUTEMPLATEA HGLOBAL LPVOI  
D  
RPC\_IF\_HANDLE\_X HOOKPROC  
45 \_p\_ PSTR\_z LPCFHOOKPROC !\_ USHORT\_7\_



LPHMIDIOUT\_\_HRESULT\_\_  
\_\_PUCHAR\_\_DISPID\_\_È\_\_PFULL\_PTR\_XLAT\_TABLES\_\_ž\_LPCCHOOKPROC\_\_  
\_\_LPBYTE\_\_  
\_\_X\_\_PROC  
5 \_\_A\_\_DATE\_\_D\_\_  
ENHMFENUMPROC  
\_\_PSID\_\_  
\_\_t\_\_LPINT\_\_  
\_\_N\_\_SNB  
10 \_\_p\_\_CHAR\_\_p\_\_LPCSTR\_\_Ó\_\_SERVER\_ROUTINE\_\_  
\_\_SCODE&\_\_Ø\_\_LPOVERLAPPED\_COMPLETION\_ROUTINE\_\_Û\_\_  
YIELDPROC\_\_!\_\_VARTYPE\_\_HWAWEIN\_\_ß\_\_IRpcChannelBuffer\_\_ß\_\_IRpc  
ChannelBuffer\_\_ž\_\_LPPAGEPAINTHOOK\_\_t\_\_  
OLERENDER\_\_u\_\_LPUINT\_\_X\_\_GOBJENUMPROC\_\_  
15 HMETAFILEPICT\_\_N\_\_LPBSTR\_\_  
\_\_byte\_\_@\_\_PFLOAT\_\_"\_\_PSECURITY\_INFORMATION\_\_\$\_\_NDR\_RUNDOWN\_\_  
LPCVOID\_\_"\_\_SERVICE\_STATUS\_HANDLE\_\_ž\_\_LPPAGESETUPHOOK\_\_  
\_\_u\_\_u\_\_int\_\_!\_\_wint\_\_t\_\_â\_\_  
20 LPMMIOPROC\_\_  
STUB\_THUNK\_\_  
\_\_9\_\_sval\_\_  
\_\_!\_\_WCHAR\_\_HGDIOBJ\_\_X\_\_ENUMRESNAMEPROC\_\_HCONVLIST\_\_  
\_\_HSZ\_\_  
25 \_\_t\_\_INT\_\_HDDEDATA\_\_å\_\_tagHANDLETABLE\_\_X\_\_FARPROC\_\_X\_\_  
PROPENUMPROCW\_\_"\_\_MCIERROR\_\_t\_\_L\_\_€\_\_€\_\_  
\_\_\_\_\_(())\*\*,-  
..//1122334455668899::;<<==>>@@AABBCCDDEEFFGGHHIIJJKKLLMMNNOOPPQQ  
30 RRSSTTUUVVWWXXYYZZ[[\]]^`ddeehhiikkllmmooqqssttwwxx{||}}€€□□ff,,,†  
†^%o%oSŠ<<€€□□ŽŽ□□''''''..\_\_\_\_  
TMTM©©aa««««®®°°±±2233''µµ¶¶··.ºº»»¼¼½½ÀÀÁÁÂÂÃÃÄÄÅÅÆÆÇÇÈÈÉÉÊÊËËÌÌÍÍÎÎÏÏÐÐÓÓÔÔÕÕÖÖ××ØØÙÙÚÚÛÛ  
ääççééêêëëììííîîïïððññ~<€\_@€8:€IB€,7€i<€\<€xBe,5€h3€-9€,@€x4€@=€H5€i3€\_3€Ä>€ÀA€8  
@€i>€@€,,@€4€ðA€:€Û@€\_@€Ä;€\$:€`6€I@€"7€ð6€i7€ð2€hA€Ä<€i=€`A€\_A€,?€~A€\_3€  
35 XA€p?€¼9€\$9€HA€">€,,>€@7€à?€ð?€T@€¼@€8A€H;€È2€L6€P7€°8€X3€\_@€ð3€Û4€t5  
€ø:€Ö>€\$>€ø9€I€C(A€|3€\_4€t8€9€€45€□3€D@€ø7€PCE¼B€,3€'6€X=€`5€'>€`4€,<€ø>€\_?  
€Û3€i;€d7€"=€-@€□7€`5€t@€\;€p;€D9€H<€d@€\_,;€B€,5€":€p6€<€\_;€T9€üB€,<9€äA€äC€  
\_6€i9€ä9€Ø;€Û8€°<€|C€°;€\_4€h:€\$C€<€ðB€,6€i=€€=€€9€0;€"=€7€È4€ø8€-B€,9€È6€t>  
\_€ø8€È5€`6€\_?€øA€\_€D>€ä:€4>€A€d>€XB€X?€T>€`A€\_C€,7€<B€,ø3€¼C€,À=€";€8C€I  
40 C€C€4€p<€,B€D8€,8€xA€,8€ü@€\8€i@€P:€x7€À:€\_5€8?€øø6€(?€D?€>€□C€,0<€=€P4€Û5  
\_€DA€\$4€ØB€,?€D3€,5€,3€Û2€,C€,B€,C€,B€,Ä8€Ø=€|:€-C€,8€ð5€,€:€□@€<4€È7€,ë\_\_aut  
odr\_\_breake\_\_case\_\_R\_\_charl\_\_consthk\_\_continuei9\_\_defaultky\_\_do\_\_X\_\_doublemQ\_\_else  
\_\_,\_\_enum\_\_  
\_È\_\_extern\_\_°\_\_floatq\_\_fors\_\_gotot\_\_  
45 \_\_if\_\_À\_\_int\_\_0\_\_long\_\_register\_\_ù\_\_return\_\_ÿ\_\_short\_\_

```

signed/m_sizeof    static T_struct†ø_switch_I_typedef
union - unsigned > void2: volatile%oy while ' inline(ó cdecl. based,,
I stdcallÄ< declspec±Lasm²& catch classðÄ const cast□^deleteñĬ dynamic_cas
t_çfriend_n_€inlineŽnew™¢_operator<I_private€©_protected□»_publicēûreinterpret_castĬ
5 static_cast³Étemplate¹_this'&throwµ..._try{ð_typeid_&_virtualĭ—
namespaceĭH_using >_single_inheritance_û_multiple_inheritance_____virtual
inheritance_$_novtordisp;_resume!_nounwind_~_syscall_r_near_
_Á_far_____far16_ý_huge_-_fortran_d4€
export_ð_interrupt_s_loadds_a_saveregs_Ä_segment²_segname_Ö_self_
10 ^_thiscall(û_cdecl_Y_far`_fortran_r_huge_ç_near_4_pascal_Ô_pascal©ĭ_
_try^_except«“_finally@_leave-
_asm)Æ_fastcall_€_unaligned_™_builtin_alignof_c_sysapi_)_b
uiltin_isfloat_±_wchar_t_”_restrictÖÖ_int8×m_int16Øq_int32Ü_ in
t64_ò_int128â#wchar_t_ë_inline(û_cdecl. based,¥_stdcallÂP_declspec_Ö
15 _syscall_w_near_1_far_far16_huge_fortran_export_.pasca
l©]_try^)_except«ĭ_finally_V_ leave-
$ _asm)†_fastcallÖÜ_int8×f_int16Ø†_int32Ü•_int64*_€dllimport_Ø_€dllexport³_
€naked_3_€thread_·_€allocate6_strset2_setjmpd_rotl^_rotr%_outpww_outpd_outp
~_lrotrx_lrotr_inpwð_inpd”_inp/_exception_info_exception_codeQ_enable_disa
20 ble^_alloca”_abnormal_termination,_ull_rshiftØ<€_ll_rshiftE_ll_lshiftb_emuluû_
_emul+_ReturnAddressð4€_atan2Ž_atanž_asin»acos_abs_cosh~?€bcos□exp_fmod$ fabsx
_inpw_e_inpd¼_inpi_log1086€çlog_labsÄ_memsetômemcpyßmemcmp%_outpww_outpdr_ou
tp7_pow<_strset/€_strlenl_strcpyW_strcmp+_strcat_sqrt'_sinhF_sinR_tanh_>€6_tanždefine
d_____formalÇ_...Äabstract_declaratorf_main†_WinMain_DllMain»_alignment
25 member.z_unnamed,=€$€C.drectve□_local_static
guard.ðCatch@_LOCAL_SIZE!_LINE_Var]_LINE_'_FILE_¥_DATE_
M_TIME_ù_TIMESTAMP_”_INTEGRAL_MAX_BITSú_M_IX86
_MSC_VER,_WIN32{ _MT¹_WIN32Ĭ_NDEBUG8_WINDOWSE_J:\Vb60\Line2
1\Parity.cÄžEÜ{€_€P{€{€_€x{€iz€_Ĭð{€,Ĭ€{€Øžed{€<{€'_€€{€{€_Ĭ€p-
30 €Ø'€”žEÖ'€_žE_□€h'€È'€ä□€
□€”€t□€”žEøžE,žEA@cc@ABB@CC@LL@DD@_€A_A@c_'€|€x€€¼¼€H_Ex|€
ÿ_”€.”€_€ä”€Ĭ€4”€Ä•€Ü_€_€°_€D”€4□€h”€D™€€—
€Ĭ'€r€”€_™€<€_”€_”€□€H□€_€_—€ð□€€□€_□€Ĭ'€_□€Ü_€Ä_€_”€<•€,-€ä—
€€”€”_”€<€_□€_—€4_€_—€_•€@'€_”€œ_€ä□€_□€è_€p”€D-
35 €”_€h_€Ĭ_€_Ĭ_xžEç_žE_€_žE_>€□€_£_—žE_xžEA_üžE_,>€_žE_Ü_□_<€x□€_
c@_žE_l>€x□€_Ÿ_\>€ä□€_°_B@_žE_l>€$□€_á_D@_žE_l>€žE_©_€>€$□€_
_ç_œ>€$□€_Ĭ_—>€ä□€_¼>€$□€_à_Ĭ>€žE_”_Ü>€_Ĭ'€_Ú€_üžE_ð>€_$□
_€_Ü€_?€Ø'€_±€_žE_?€Ø'€_²_0?€_Ĭ'€
_Ø_□_@?€_žE!_×€_”€œ'€_ē_L@_žE_x?€Ü'€#_ē_L_žE_€?€œ'€$_ē_?€œ'€%_Ĭ_°?
40 €Ü'€&_Ĭ_Ä?€_Ĭ'€_ä€_Ø?€Ĭ'€( ..._žE_è?€Ĭ'€)_f_ø?€Ĭ'€*_ _□_@€Ĭ'€+_ _□_@€”
žE,_ _$@€Ĭ'€-
_¥_ _“€4@€Ĭ'€_o_@@€Ĭ'€/_a_L@€Ĭ'€0_ _\@€Ĭ'€1_ _Ĭ@€ä□€2_ _A_Ĭ@€$□€3_à
A_€@€”žE4_”A_~@€Ĭ'€5_Ĭ_”@€Ĭ'€6_e_’@€Ĭ-
€7_ _D_žE_Ä@€Ĭ'€8_ _Ö@€Ĭ'€9_ _ä@€žE: _a_ð@€ä□€_°B_ _A€x□€<_áb_ _A€ž
45 €=_©B_$A€Ĭ'€>_®_0A€□€?_£B_@A€x□€@_$_ _PA€□€A_□_”A€žEB_¥_pA€□€C_Ĭ_
_€A€Ĭ'€D_]_□_A€Ĭ'€E_‡_ _A€Ĭ'€F_k_ _A€Ĭ'€G_<_ĬžE¼A€Ĭ'€H_s_ _ÈžE_ _ÈžEP

```

5



File: stable.obj

```

L_
5  mTM9d_-.drectve(d
   _datax_œ_p@@À-defaultlib:LIBCMT -defaultlib:OLDNAMES ,__
   +
   *_)_$_($_)4
   8&@_D"_H!_P_T(X_`
10  _h_p_t+_
   €*_,)œ_□$")œ
   &"_~"!_°!_¼_À_È
   Ì_Ð_Ø
   Û_à_è
15  Ì_ð_ø_ü
      (
      ,_0_4_<_
      @_D_L_
20  P_(X_
   `_,h_1(p'x_€_,^
   œ(□#~&"_~*°)_¼_$_À_)È
   Ì"Ð!Ø_Û_(à_è_ð(ô_%ü
   ,_+_*_)$_($_)4
25  _8&@_D"_H!_P_T_X_\_d_h_(l_t_.filepÿg_J:\Vb60\Line21\stable.c
   .drectve_(.data_x_p_ð_€_;'_Jh_X_XDSDData_c_s_CRTest
   _PACTest_,_"_®_,_Â_Î_URLTest_Û_è_ð_'_4_XDSCont_A_
   _XDSEnd_N_Z_d_q_{_%_""
   ž_URLDataState_XDSEndState_XDSDDataState_XDSContState_XDSSStartState_ControlSt
30  ate_DataState_CtrlChangeTest_ChannelChange_EndOfCaptionTest_TextDecode_TextMode
   Test_DumpText_DataSave_URLEndTest_URLDecode_PaintOnDecode_PaintOnTest_RollU
   pDecode_RollUpTest_PopOnDecode_PopOnTest_ControlSave_ControlTest_XDSContTest_
   XDSEndTest_XDSSStart_DefaultTest_XDSBegin_XDSSStartTest_F1XDSTest_NullTest

35

```

File: Tests.obj

```
L_#_mTM9e
5 Z.drectve(Œ_
  .text'_P'.debug$F_°Ê_H_B.text
  Œ_P'.debug$F_þî_H_B.text_ø_P'.debug$F_____H_B.text_)
  P'.debug$F_B_R_H_B.text\_P'.debug$F_u..._H_B.text
  □_P'.debug$F_œ¬_H_B.text_¶_P'.debug$F_Ê_Û_H_B.text5â_
10 P'.debug$F_*_H_B.text54_P'.debug$F_i_y_H_B.text0f_
  P'.debug$F_³_Ä_H_B.text0Í_P'.debug$F_ý_
  H_B.text0_P'.debug$F_G_W_H_B.text&a_P'.debug$F_‡_—_H_B.text;i_
  P'.debug$F_Û_i_H_B.text0ö_P'.debug$F_&_6_H_B.text+@
  P'.debug$F_k_{_H_B.textÆ..._P'.debug$F_K
15 [
  H_B-defaultlib:LIBCMT -defaultlib:OLDNAMES ,_Ä____€|$__Ä÷ØÄ
  €|$_s
  f|$_u_,_Ä3ÄÄ__ŠD$_,_Ät<_s
  "t_,_Ä3ÄÄ__ŠD$_,_Ät<_s
20 "u_,_Ä3ÄÄ__ŠD$_,<_Ä÷ØÄ
  __ŠD$<_v
  <s_,_Ä3ÄÄ_%_<D$_<L$_<D□üfø_t_fø_t
  fø_t_fø_uŠD$<*_t<+t_3ÄÄ_,_Ä5_*_<D$_<L$_<D□üfø_t_fø_t
  fø_t_fø_u_ŠD$<$v
25 <(s_,_Ä3ÄÄ5/_<L$_<D$_<D`üfø_t_fø_t
  fø_t_fø_u
  €|$_)u_,_Ä3ÄÄ0_4_<D$_<L$_<D□üfø_t_fø_t
  fø_t_fø_u
  €|$_u_,_Ä3ÄÄ0_9_<D$_<L$_<D□üfø_t_fø_t
30 fø_t_fø_u
  €|$/_u_,_Ä3ÄÄ0_>_<D$_<L$_<D□üfø_~_fø_}
  €|$_?v_,_Ä3ÄÄ&_C_<L$_<D$__ŠT$__□D`ü€ú_<H_<v_€ú@s_fø_}...Éu
  fø_~_fù_t_,_Ä3ÄÄ;_H_<D$_<L$_<D□üfø_t_fø_t
  fø_t_fø_u
35 €|$_-
  u_,_Ä3ÄÄ0_M_<L$_<D$__fù_<D`üu_fø_u_ŠD$<*_t<+u_,_Ä3ÄÄ+_R_<L$_<D$__I<ÑÄÄ_Ä
  á
  <'$_fú_□_,_Ž_€|$_]...□_Ä€xú[...,,ŠHü€ù0|_€ù9~_€ùA|_€ùF~
  €ù|h€ùf_□cŠHy€ù0|_€ù9~_€ùA|_€ùF~
40 €ù|h€ùf_□BŠHp€ù0|_€ù9~_€ùA|_€ùF~
  €ù|&€ùf_□!Š@ÿ<0|_<9~_<A|_<F~_<a|
  <f_□_,_Ä3ÄÄÆ_W_.fileþÿg_J:\Vb60\Line21\Tests.c.drectve__(.text_____
  .debug$F_____.text_____
  _____debug$F_____.text_____debug$F_____.text_____&__.debug$F
45 _____text
  _____4
```



File: XDCMessages.obj

```

L_m™9ê :directve(ä_
5 _text_____P'.data_9_@_0À.data_<_@_0À.debug$F_A_Q_H_B.textý[_X_

_P'.data_Ú_@_0À.data_i_@_0À.data_ï_@_0À.data_ò_@_0À.data_õ_@_0À.data_ø_@_0À
.data
ü_@_0À.data
10 _@_0À.debug$F_____H_B.texth_*'______P'.data_Î_@_0À.debug$F_Ð_à
_H_B-defaultlib:LIBCMT -defaultlib:OLDNAMES <
SV,<TSW\$_U%o
Š@_`B_<fH_Áà_fL_□@+À□<M'ÿÿÿÿò@÷Ñ+ù<é<÷'ÿÿÿÿú+Àò@O<ÍÁé_ó¥Ífá_ó¤¿'ÿÿÿ
ÿ+Àò@÷Ñ+ù<é<÷'ÿÿÿÿú+Àò@O<ÍÁé_ó¥Ífá_ó¤<fH_Áà_fL_Áà_□_@<Œ_T_ÊÆ,,X_<f
15 H_Áà_fL_Áà_□@+À□¼_X_'ÿÿÿÿò@÷Ñ+ù<Û<÷'ÿÿÿÿú+Àò@O<ÊÁé_ó¥Ê]fá_ó¤^[Ä_
_6_d
_XDS
_____fi_,S<_VW<T$ _Ud$_%MŠ@_`E_<,H_Áà_,L_□@+À□<M'ÿÿÿÿò@÷Ñ+ù<Û<÷'ÿÿÿ
20 ÿÿ<ý+Àò@O<ÊÁé_ó¥Êfá_ó¤<,H_Áà_,L_Áà_□@Šœ_X_□_Š,,Y_`D$_ŠÄ$_Äè_,Ä_,ð_
<_,è<_..._'ÿÿÿÿ+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Ê3Òfá_ó¤ŠT$_'ÿÿÿÿfá_+
Ä<<÷ò@÷Ñ+ù%L$_<÷'ÿÿÿÿý+Àò@O<L$_Áé_ó¥L$_€ä_Äè_fá_ó¤ŠD$_$8
ØŠD$_Äü_$
ØÄü_"šöÄ_t0¿'ÿÿÿÿ+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Êfá_ó¤öÄ_t0¿'ÿÿÿÿ+Àò@÷
25 Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Êfá_ó¤öÄ_t0¿'ÿÿÿÿ+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁ
é_ó¥Êfá_ó¤öÄ_t0¿'ÿÿÿÿ+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Êfá_ó¤öÄ_,Ü¿'ÿÿÿÿ+
Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Ê]fá_ó¤^[fÄ_Ä¿'ÿÿÿÿ+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+
Àò@O<ÊÁé_ó¥Ê]fá_ó¤^[fÄ_Ä¿'ÿÿÿÿ+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Êfá_ó¤3
Ä'ÿÿÿÿŠÄfä_<...+Àò@÷Ñ+ù<Ñ<÷'ÿÿÿÿý+Àò@O<ÊÁé_ó¥Êfá_ó¤]^[fÄ_Ä_
30 _6_-(i.;-E%_z_"_ä_____U_□_+_Ë_,_OTHER - non-USA,D,L,S,V,FV
TVPG - MPAA -
ý_____<_SV,<LSWU%o_Š@_<T$_`A_<,H_Áà_,L_□@+À□<M'ÿÿÿÿò@÷Ñ+ù<Û<÷'ÿÿÿ
<|$_+Àò@O<ÊÁé_ó¥Êfá_ó¤¿'ÿÿÿÿ+Àò@÷Ñ+ù<Û<÷'ÿÿÿÿ<|$_+Àò@O<ÊÁé_ó¥Êfá_ó¤3Ü<,
H_Áà_,L_Áà_□@9œ_T_Ž<,H_Áà_,L_Áà_□_@_Ä_¼Œ_X_+Ä<<□€ÿÿÿÿ'ÿÿÿÿò@÷Ñ+
35 ù<é<÷'ÿÿÿÿ<|$_+Àò@O<ÍÁé_ó¥Ífá_ó¤¿'ÿÿÿÿ+Àò@÷Ñ+ù<é<÷'ÿÿÿÿ<|$_+Àò@O<ÍÁé_Có¥Íf
á_ó¤<,H_Áà_,L_Áà_□@9œ_T_□Xÿÿÿ]_[Ä_
_6_f
_ß7_6_h_3_.fileþÿg_J:\Vb60\Line21\XDSMessages.c.directve_(.text_____Def_
40 _data_____data_____XDSMes_.debug$F_____text_ý_
_Vcp__.data_3_.data_____data_u_.data
____<
_.data_____i_.data_____:.data
45 _Î_
_.data_____

```

2025-09-24 14:54:54



5     \_i\_\_\_\_\_'.debug\$F\_\_\_\_\_ .text \_\_\_\_h\_\_\_\_\_Typ\_  
      \_.data\_\_\_\_\_4\_\_\_\_\_types\_.debug\$F\_\_\_\_\_I ??\_C@\_02IHLC@?5?5?\$AA@??\_C@\_04EPN  
      H@XDS?5?\$AA@??\_C@\_0BC@NEMK@?5?5OTHER?5?9?5non?9USA?\$AA@??\_C@\_  
      02DFDA@?0D?\$AA@??\_C@\_02GBII@?0L?\$AA@??\_C@\_02MJKF@?0S?\$AA@??\_C@\_  
      @\_02MJGO@?0V?\$AA@??\_C@\_03LEKC@?0FV?\$AA@??\_C@\_09GMHB@?5?5TVPG  
      ?5?9?5?\$AA@??\_C@\_09NPEP@?5?5MPAA?5?9?5?\$AA@\_mpaaRatings\_flagmask\_tvp  
      gRatings??\_C@\_01FCOA@?5?\$AA@

$$\underline{L_v}$$

5 m™9¶io\_directve(,,  
 .data \_ @\_0À.data ' @\_0À.data ¼ @\_0À.data Ä @\_0À.data Ì @\_0À.data Ô @\_0À.  
 data Û @\_0À.data ä @\_0À.data ê @\_0À.data ò @\_0À.data ù @\_0À.data  
 @\_0À.data  
 @\_0À.data @\_0À.data @\_0À.data @\_0À.data  
 10 + @\_0À.data 4 @\_0À.data < @\_0À.data C @\_0À.data K @\_0À.data S @\_0À.  
 .data Z @\_0À.data a @\_0À.data h @\_0À.data p @\_0À.data w @\_0À.data  
 ~ @\_0À.data < @\_0À.data“ @\_0À.data Ÿ @\_0À.data  
 “ @\_0À.data ± @\_0À.data , @\_0À.data  
 ÿ @\_0À.data È @\_0À.data Ð @\_0À.data Ö @\_0À.data  
 15 á @\_0À.data ê @\_0À.data ò @\_0À.data ú @\_0À.data ÿ @\_0À.data \_\_ @\_0À.d  
 ata\_  
 @\_0À.data @\_0À.data  
 @\_0À.data # @\_0À.data1 @\_0À.data= @\_0À.data I @\_0À.data P @\_0À.data U @  
 \_0À.data \ @\_0À.data b @\_0À.dataj @\_0À.data v @\_0À.data } @\_0À.data ^ @\_0À.  
 20 data □ @\_0À.data  
 ” @\_0À.dataž @\_0À.data ^ @\_0À.data  
 ^ @\_0À.data » @\_0À.data À @\_0À.data È @\_0À.data Ð @\_0À.data Ö @\_0À.  
 .data  
 Ý @\_0À.data æ @\_0À.data î @\_0À.data ù @\_0À.dataÿ @\_0À.data \_\_ @\_0À.d  
 25 ata \_\_ @\_0À.data  
 @\_0À.data \$ @\_0À.data -  
 @\_0À.data 5 @\_0À.data @\_ @\_0À.data G @\_0À.data N @\_0À.data  
 V @\_0À.data ` @\_0À.data i @\_0À.data r @\_0À.data  
 } @\_0À.data † @\_0À.data □ @\_0À.data  
 30 ~ @\_0À.data ¢ @\_0À.data « @\_0À.data ´ @\_0À.data À @\_0À.data Æ @\_0À.data  
 Í @\_0À.data × @\_0À.data Û @\_0À.data â @\_0À.data  
 ð @\_0À.data ú @\_0À.data \_\_ @\_0À.data \_\_ @\_0À.data \_\_ @\_0À.data \_\_  
 @\_0À.data \_\_ @\_0À.data  
 % @\_0À.data / @\_0À.data 1 @\_0À.data 7 @\_0À.data 9 @\_0À.data ? @\_0À.data B  
 35 @\_0À.data D @\_0À.dataÜ+H\_\$Bö @@À-defaultlib:LIBCMT -defaultlib:OLDNAMES  
 Roll4 Roll3 Roll2 Paint Pop  
 WesternWeatherVideoVarietyTravelTennisTechnicalTalkSuspenseSpecialSoap  
 OperaShoppingServiceSeriesScienceRomanceReviewRepeatRepairReadingRacingPublicProf  
 40 essionalProductPrerecordedPremierePoliticsPoliceNatureNationalMysteryMusicMiniseriesM  
 ilitaryMeetingMedicalMathLocalLiveLegalLanguageInterviewInternationalInstructionInform  
 ationHorrorHomeHockeyHobbyHistoryHigh  
 SchoolHealthGovernmentGolfGardenGame/QuizFund  
 RaiserForeignFootballFoodFictionFashionFarmFantasyExerciseEroticaElementaryDramaDoc  
 umentaryDanceCrimeContemporaryConsumerConcertCommentaryComedyCombatCollegeC  
 45 lassicalBusinessBulletinBasketballBaseballAwardsAutomobileAnthologyAnimatedAdvertise  
 mentActionOTHERSportsReligiousNewsMovieEntertainmentEducationTV-MA TV-14 TV-

PG TV-GTV-Y7 TV-YNNoneNot RatedXNC-17RPG-13PGGN/A\_\_ \_ \_

5

10

15

Null StartDuratShow Type

VChipAudioCapti0108 Aspec010A 010B Comp1Comp2010E 010F

Desc1Desc2Desc3Desc4Desc5Desc6Desc7Desc80118 0119 011A 011B 011C 011D 011E

011F 0120 0121 0122 0123 0124 0125 0126 0127 0128 0129 012A 012B 012C 012D 012E

012F 0130 0131 0132 0133 0134 0135 0136 0137 0138 0139 013A 013B 013C 013D 013E

20

013F 0140 0141 0142 0143 0144 0145 0146 0147 0148 0149 014A 014B 014C 014D 014E

014F 0150 0151 0152 0153 0154 0155 0156 0157 0158 0159 015A 015B 015C 015D 015E

015F 0160 0161 0162 0163 0164 0165 0166 0167 0168 0169 016A 016B 016C 016D 016E

016F 0170 0171 0172 0173 0174 0175 0176 0177 0178 0179 017A 017B 017C 017D 017E

017F 0300 0301 0302 0303 0304 0305 0306 0307 0308 0309 030A 030B 030C 030D 030E

25

030F 0310 0311 0312 0313 0314 0315 0316 0317 0318 0319 031A 031B 031C 031D 031E

031F 0320 0321 0322 0323 0324 0325 0326 0327 0328 0329 032A 032B 032C 032D 032E

032F 0330 0331 0332 0333 0334 0335 0336 0337 0338 0339 033A 033B 033C 033D 033E

033F 0340 0341 0342 0343 0344 0345 0346 0347 0348 0349 034A 034B 034C 034D 034E

034F 0350 0351 0352 0353 0354 0355 0356 0357 0358 0359 035A 035B 035C 035D 035E

30

035F 0360 0361 0362 0363 0364 0365 0366 0367 0368 0369 036A 036B 036C 036D 036E

036F 0370 0371 0372 0373 0374 0375 0376 0377 0378 0379 037A 037B 037C 037D 037E

037F 0500 NetwkAffilDelay0504 0505 0506 0507 0508 0509 050A 050B 050C 050D 050E

050F 0510 0511 0512 0513 0514 0515 0516 0517 0518 0519 051A 051B 051C 051D 051E

051F 0520 0521 0522 0523 0524 0525 0526 0527 0528 0529 052A 052B 052C 052D 052E

35

052F 0530 0531 0532 0533 0534 0535 0536 0537 0538 0539 053A 053B 053C 053D 053E

053F 0540 0541 0542 0543 0544 0545 0546 0547 0548 0549 054A 054B 054C 054D 054E

054F 0550 0551 0552 0553 0554 0555 0556 0557 0558 0559 055A 055B 055C 055D 055E

055F 0560 0561 0562 0563 0564 0565 0566 0567 0568 0569 056A 056B 056C 056D 056E

056F 0570 0571 0572 0573 0574 0575 0576 0577 0578 0579 057A 057B 057C 057D 057E

40

057F 0700 Time ImpulSupplZone 0705 0706 0707 0708 0709 070A 070B 070C 070D 070E

070F 0710 0711 0712 0713 0714 0715 0716 0717 0718 0719 071A 071B 071C 071D 071E

071F 0720 0721 0722 0723 0724 0725 0726 0727 0728 0729 072A 072B 072C 072D 072E

072F 0730 0731 0732 0733 0734 0735 0736 0737 0738 0739 073A 073B 073C 073D 073E

073F OOBNB0741 0742 0743 0744 0745 0746 0747 0748 0749 074A 074B 074C 074D

45

074E 074F 0750 0751 0752 0753 0754 0755 0756 0757 0758 0759 075A 075B 075C 075D

075E 075F 0760 0761 0762 0763 0764 0765 0766 0767 0768 0769 076A 076B 076C 076D

PG TV-GTV-Y7 TV-YNNoneNot RatedXNC-17RPG-13PGGN/A\_\_ \_ \_

076E 076F 0770 0771 0772 0773 0774 0775 0776 0777 0778 0779 077A 077B 077C 077D  
077E 077F 0900 0901 0902 0903 0904 0905 0906 0907 0908 0909 090A 090B 090C 090D  
090E 090F 0910 0911 0912 0913 0914 0915 0916 0917 0918 0919 091A 091B 091C 091D  
091E 091F 0920 0921 0922 0923 0924 0925 0926 0927 0928 0929 092A 092B 092C 092D  
5 092E 092F 0930 0931 0932 0933 0934 0935 0936 0937 0938 0939 093A 093B 093C 093D  
093E 093F 0940 0941 0942 0943 0944 0945 0946 0947 0948 0949 094A 094B 094C 094D  
094E 094F 0950 0951 0952 0953 0954 0955 0956 0957 0958 0959 095A 095B 095C 095D  
095E 095F 0960 0961 0962 0963 0964 0965 0966 0967 0968 0969 096A 096B 096C 096D  
096E 096F 0970 0971 0972 0973 0974 0975 0976 0977 0978 0979 097A 097B 097C 097D  
10 097E 097F 0B00 WSCodWSMes0B03 0B04 0B05 0B06 0B07 0B08 0B09 0B0A 0B0B  
0B0C 0B0D 0B0E 0B0F 0B10 0B11 0B12 0B13 0B14 0B15 0B16 0B17 0B18 0B19 0B1A  
0B1B 0B1C 0B1D 0B1E 0B1F 0B20 0B21 0B22 0B23 0B24 0B25 0B26 0B27 0B28 0B29  
0B2A 0B2B 0B2C 0B2D 0B2E 0B2F 0B30 0B31 0B32 0B33 0B34 0B35 0B36 0B37 0B38  
0B39 0B3A 0B3B 0B3C 0B3D 0B3E 0B3F 0B40 0B41 0B42 0B43 0B44 0B45 0B46 0B47  
15 0B48 0B49 0B4A 0B4B 0B4C 0B4D 0B4E 0B4F 0B50 0B51 0B52 0B53 0B54 0B55 0B56  
0B57 0B58 0B59 0B5A 0B5B 0B5C 0B5D 0B5E 0B5F 0B60 0B61 0B62 0B63 0B64 0B65  
0B66 0B67 0B68 0B69 0B6A 0B6B 0B6C 0B6D 0B6E 0B6F 0B70 0B71 0B72 0B73 0B74  
0B75 0B76 0B77 0B78 0B79 0B7A 0B7B 0B7C 0B7D 0B7E 0B7F 0D00 0D01 0D02 0D03  
0D04 0D05 0D06 0D07 0D08 0D09 0D0A 0D0B 0D0C 0D0D 0D0E 0D0F 0D10 0D11  
20 0D12 0D13 0D14 0D15 0D16 0D17 0D18 0D19 0D1A 0D1B 0D1C 0D1D 0D1E 0D1F  
0D20 0D21 0D22 0D23 0D24 0D25 0D26 0D27 0D28 0D29 0D2A 0D2B 0D2C 0D2D  
0D2E 0D2F 0D30 0D31 0D32 0D33 0D34 0D35 0D36 0D37 0D38 0D39 0D3A 0D3B 0D3C  
0D3D 0D3E 0D3F 0D40 0D41 0D42 0D43 0D44 0D45 0D46 0D47 0D48 0D49 0D4A 0D4B  
0D4C 0D4D 0D4E 0D4F 0D50 0D51 0D52 0D53 0D54 0D55 0D56 0D57 0D58 0D59 0D5A  
25 0D5B 0D5C 0D5D 0D5E 0D5F 0D60 0D61 0D62 0D63 0D64 0D65 0D66 0D67 0D68 0D69  
0D6A 0D6B 0D6C 0D6D 0D6E 0D6F 0D70 0D71 0D72 0D73 0D74 0D75 0D76 0D77  
0D78 0D79 0D7A 0D7B 0D7C 0D7D 0D7E 0D7F  
k k k k j i k k  
k \$ k ( k , k 0 k 4 k 8 k < k @ k D k H k L k P k T k X k  
30 \ k ` k d k h k l k p k t k x k | k € k „ k ^ k Æ k □ k ” k ~  
k œ k k □ k “ k ¬ k ° k ´ k , k ¼ k Å k Ä k Ë k Ì k Ð k  
Ô k Ø k Û k à k ä k è k ì k ð k ô k ø k ü k k k k k  
k k k k  
k \$ k ( k , k 0 k 4 k 8 k < k @ k D k H k L k P k T k X k  
35 \ k ` k d k h k l k p k t k x k | k € k „ k ^ k Æ k □ k ” k ~  
k œ k k □ k “ k ¬ k ° k ´ k , k ¼ k Å k Ä k Ë k Ì k Ð k  
Ô k Ø k Û k à k ä k è k ì k ð k ô k ø k ü k k k k k  
k k k k  
k \$ k ( k , k 0 k 4 k 8 k < k @ k D k H k L k P k T k X k  
40 \ k ` k d k h k l k p k t k x k | k € k „ k ^ k Æ k □ k ” k ~  
k œ k k □ k “ k ¬ k ° k ´ k , k ¼ k Å k Ä k Ë k Ì k Ð k  
Ô k Ø k Û k à k ä k è k ì k ð k ô k ø k ü k k k k k  
k k k k  
k \$ k ( k , k 0 k 4 k 8 k < k @ k D k H k L k P k T k X k  
45 \ k ` k d k h k l k p k t k x k | k € k „ k ^ k Æ k □ k ” k ~  
k œ k k □ k “ k ¬ k ° k ´ k , k ¼ k Å k Ä k Ë k Ì k Ð k



à)k ä)k è)k i)k ð)k ô)k ø)k ü)k \*` \*] \*Z \*W \*T \*Q \*N \*  
 K \*H \$\*E (\*B \*? 0\* < 4\*9 8\*6 <\*H H\*3 L\*0 P\*  
 T\*\* X\* '\\*\$ '! d\* h\* l\* p\* t\* x\* |\* €\*  
 \* ^\* Æ\* □\*ý \*\*ú ~\*÷ œ\*ô \*ñ □\*î \*\*ë -\*è °\*â ' \*â \*ß ¼\*Ü À\*Ü  
 5 Ä\*Ö È\*Ö Î\*Ð Ð\*Î Ô\*Ê Ø\*Ç Ü\*Ä à\*Á ä\*¼ è\*» i\* \_ð\*µ ð\*² ø\* \_ü\*~ +© \_+! \_+£  
 + \_+□ \_+š \_+ \_+”  
 +‘ \$+Ž (+< \_+ 0+... 4+ \_+ 8+ 𐀀 <+| @+y D+v H+s L+p P+m T+j X+g \+d `+a d+^ h  
 +[ l+X p+U t+R x+O |+L €+H \_+F ^+C Æ+@ □+= \_+”: ~+7 œ+4 +1 □+ \_+”++ \_+”( °  
 +% \_+” \_+ ¼+ \_+À+ \_+Ä+ \_+È+ \_+Î+ \_+Ð+  
 10 Ô+  
 Ø+ .filepÿg J:\Vb60\Line21\XDSROM.c.drectve (.data .data  
 .data < .data W .data s .data ‘ .data « .data Å  
 .data  
 Ý  
 15 .data ÷ .data .data  
 )  
 .data E .data \ .data w .data ‘ .data  
 .data Ê .data ä .data ý .data .data 1  
 20 .data J .data c .data | .data - .data .data  
 È .data è .data .data .data!  
 ;! .data" V " .data# o # .data\$  
 ^ \$ .data% £ % .data& ½ & .data' Õ ' .data(  
 ó ( .data) .data\* ( \* .data+ B + .data, Y , .data- q -  
 25 .data. ^ .data/ / .data0  
 » 0 .data1 × 1 .data2 ø 2 .data3 3 .data4 6 4 .data5 O 5 .data6  
 f 6 .data7 ~ 7 .data8 -  
 8 .data9 ° 9 .data: Ð : .data; é ; .data< < .data= = .data>  
 7 > .data? T ? .data@ t @ .dataA  
 30 Ž A .dataB © B .dataC Ä C .dataD Ú D .dataE ô E .dataF  
 F .dataG  
 % G .dataH @ H .dataI Z I .dataJ x J .dataK □ K .dataL  
 L .dataM Ç M .dataN  
 ß N .dataO  
 35 Ÿ O .dataP P .dataQ 3 Q .dataR Q R .dataS j S .dataT f  
 T .dataU  
 □ U .dataV ¹ V .dataW Ô W .dataX ï X .dataY  
 Y .dataZ ( Z .data[ A [ .data\  
 \ .data] { } .data^ ^ .data` ` .data` Ð` .dataa è  
 40 a .datab  
 b .datac  
 c .datad 3  
 d .datae K  
 45 e .dataf  
 1

f\_.datag\_\_\_\_^  
g\_.datah\_\_\_\_£  
h\_.datai\_\_\_\_¾  
i\_.dataj\_\_\_\_Û  
5 j\_.datak\_\_\_\_ï  
k\_.datal\_\_\_\_  
l\_.datam\_\_\_\_!\_m\_.datan\_\_\_\_  
\_7\_n\_.datao\_\_\_\_T\_o\_.datap\_\_\_\_h\_p\_.dataq\_\_\_\_□\_q\_.datar\_\_\_\_•\_r\_.datas\_\_\_\_®\_s\_.datat\_\_\_\_  
\_Ä\_t\_.datau\_\_\_\_×\_u\_.datav\_\_\_\_Û+ð\_CTypeE+v\_\_typesH\*v\_i\_@\*v\_ø\_  
10 \*v\_\_\*v\_\_XDSDisp\_v\_\_Vcp\_\_Typ\_\_Def  
XDSMes\_v\_\_€\_v\_\_v\_\_(??\_C@\_07OJHM@Roll4?5?5?\$AA@??\_C@\_07FHBN@Roll3?5  
?5?\$AA@??\_C@\_07FAL@Roll2?5?5?\$AA@??\_C@\_07ILCI@Paint?5?5?\$AA@??\_C@\_0  
7LFNA@Pop?5?5?5?5?\$AA@??\_C@\_07PCEP@Western?\$AA@??\_C@\_07GGMB@Weat  
her?\$AA@??\_C@\_05HAHE@Video?\$AA@??\_C@\_07GOLO@Variety?\$AA@??\_C@\_06  
15 MAKO@Travel?\$AA@??\_C@\_06IECD@Tennis?\$AA@??\_C@\_09MAPH@Technical?\$A  
A@??\_C@\_04MNKN@Talk?\$AA@??\_C@\_08EABE@Suspense?\$AA@??\_C@\_07GALJ  
@Special?\$AA@??\_C@\_0L@NHB@Soap?5Opera?\$AA@??\_C@\_08HFCK@Shopping?\$  
AA@??\_C@\_07KNHF@Service?\$AA@??\_C@\_06NHCF@Series?\$AA@??\_C@\_07LHB  
D@Science?\$AA@??\_C@\_07HHNK@Romance?\$AA@??\_C@\_06HNJB@Review?\$AA  
20 @??\_C@\_06NPIB@Repeat?\$AA@??\_C@\_06NNPN@Repair?\$AA@??\_C@\_07NCCF@R  
eading?\$AA@??\_C@\_06DPOE@Racing?\$AA@??\_C@\_06GCHD@Public?\$AA@??\_C@  
\_0N@EONA@Professional?\$AA@??\_C@\_07KKIB@Product?\$AA@??\_C@\_0M@BOC  
@Prerecorded?\$AA@??\_C@\_08JKHO@Premiere?\$AA@??\_C@\_08BGDL@Politics?\$AA  
@??\_C@\_06KJAF@Police?\$AA@??\_C@\_06BIAE@Nature?\$AA@??\_C@\_08CKMM@N  
25 ational?\$AA@??\_C@\_07GPIF@Mystery?\$AA@??\_C@\_05PHIM@Music?\$AA@??\_C@\_  
0L@IEGN@Miniseries?\$AA@??\_C@\_08LHGP@Military?\$AA@??\_C@\_07FDIO@Meeti  
ng?\$AA@??\_C@\_07HHJI@Medical?\$AA@??\_C@\_04LHAK@Math?\$AA@??\_C@\_05P  
HNP@Local?\$AA@??\_C@\_04HLML@Live?\$AA@??\_C@\_05NPLB@Legal?\$AA@??\_C  
@\_08DFDG@Language?\$AA@??\_C@\_09IAGO@Interview?\$AA@??\_C@\_0O@FJHI@I  
30 nternational?\$AA@??\_C@\_0M@PJEI@Instruction?\$AA@??\_C@\_0M@LKAF@Informati  
on?\$AA@??\_C@\_06KKNE@Horror?\$AA@??\_C@\_04OGPG@Home?\$AA@??\_C@\_06K  
GO@Hockey?\$AA@??\_C@\_05LHBI@Hobby?\$AA@??\_C@\_07JFNM@History?\$AA@?  
?\_C@\_0M@PJHA@High?5School?\$AA@??\_C@\_06MACP@Health?\$AA@??\_C@\_0L@  
JHGG@Government?\$AA@??\_C@\_04NPML@Golf?\$AA@??\_C@\_06CMMI@Garden?\$  
35 AA@??\_C@\_09CNCP@Game?1Quiz?\$AA@??\_C@\_0M@ENFA@Fund?5Raiser?\$AA@?  
?\_C@\_07JNNB@Foreign?\$AA@??\_C@\_08NPEO@Football?\$AA@??\_C@\_04JEEI@Foo  
d?\$AA@??\_C@\_07IHEG@Fiction?\$AA@??\_C@\_07MCOL@Fashion?\$AA@??\_C@\_04I  
FFI@Farm?\$AA@??\_C@\_07OCKK@Fantasy?\$AA@??\_C@\_08JEHB@Exercise?\$AA@?  
?\_C@\_07NCAH@Erotica?\$AA@??\_C@\_0L@JBPC@Elementary?\$AA@??\_C@\_05CGP  
40 O@Drama?\$AA@??\_C@\_0M@NOLC@Documentary?\$AA@??\_C@\_05FKAK@Dance?\$  
AA@??\_C@\_05HMON@Crime?\$AA@??\_C@\_0N@KIIC@Contemporary?\$AA@??\_C@  
\_08JELE@Consumer?\$AA@??\_C@\_07DLE@Concert?\$AA@??\_C@\_0L@LGND@Com  
mentary?\$AA@??\_C@\_06IGJL@Comedy?\$AA@??\_C@\_06CAIJ@Combat?\$AA@??\_C  
@\_07LGJO@College?\$AA@??\_C@\_09JFNA@Classical?\$AA@??\_C@\_08FMGD@Busin  
45 ess?\$AA@??\_C@\_08FKAB@Bulletin?\$AA@??\_C@\_0L@PKKG@Basketball?\$AA@??\_  
C@\_08GPNH@Baseball?\$AA@??\_C@\_06IPOB@Awards?\$AA@??\_C@\_0L@CFFG@A

utomobile?\$AA@??\_C@\_09IADM@Anthology?\$AA@??\_C@\_08HEBO@Animated?\$AA  
@??\_C@\_00@PINI@Advertisement?\$AA@??\_C@\_06NMFG@Action?\$AA@??\_C@\_05  
FGEM@OTHER?\$AA@??\_C@\_06BIM@Sports?\$AA@??\_C@\_09IEGM@Religious?\$A  
A@??\_C@\_04KJCI@News?\$AA@??\_C@\_05CHKH@Movie?\$AA@??\_C@\_00@JFFK  
5 @Entertainment?\$AA@??\_C@\_09HOLD@Education?\$AA@??\_C@\_06HOCB@TV?9MA  
?5?\$AA@??\_C@\_06BJIF@TV?914?5?\$AA@??\_C@\_06KCEN@TV?9PG?5?\$AA@??\_C  
@\_04HI@TV?9G?\$AA@??\_C@\_06HFLA@TV?9Y7?5?\$AA@??\_C@\_04CMC@TV?9Y?  
\$AA@??\_C@\_04COF@None?\$AA@??\_C@\_09HKPO@Not?5Rated?\$AA@??\_C@\_01FI  
AI@X?\$AA@??\_C@\_05DMNK@NC?917?\$AA@??\_C@\_01FJJO@R?\$AA@??\_C@\_05  
10 MAGO@PG?913?\$AA@??\_C@\_02DACJ@PG?\$AA@??\_C@\_01PACF@G?\$AA@??\_C  
@\_03PHGM@N?1A?\$AA@\_flagmask\_tvpgratings\_mpaaRatings\_MinLength\_MaxLength

2025 RELEASE UNDER E.O. 14176



Appendix C

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)  
All rights reserved unless otherwise noted

5

Cold Fusion

Showtime

10 File: AddComment.cfm

```
<cfif ParameterExists(Edit)>
```

```
    <cfif Len(Trim(NewComment)) GT 0>
```

15

```
        <!-- Get the max of EventID's --->
```

```
        <cfquery name="GetMax"
```

```
        datasource="VBIContent"
```

```
        username="caption"
```

20

```
        password="trewq55">
```

```
        select Max(Eventid)as maxid from ctlfile
```

```
    </cfquery>
```

```
    <cfset newEvent = '#incrementValue(GetMax.maxid)#'>
```

25

```
    <!-- Insert a new record and pass EventID --->
```

```
    <cfquery name="AddEvent"
```

```
    datasource="VBIContent"
```

30

```
    username="caption"
```

```
    password="trewq55">
```

```
    Insert into ctlfile
```

```
    (Eventid,
```

```
    Seconds,
```

35

```
    Type,
```

```
    Clipsource,
```

```
    Comment
```

```
    )
```

40

```
    Values
```

```
    (#newEvent#,
```

```
    #LTime#,
```

```
    'C',
```

```
    '#Clipsrc#',
```

45

```
    '#NewComment#'
```

```

    )
    </cfquery>

    </cfif>
5  <!-- Close Window --->
    <!-- <body onLoad="window.close();" --->
    <cfoutput>
    <cflocation url="/thankyou.cfm?LTime=#LTime#&DS=VBICContent&Clipsrc=#ClipSrc#">

10 </cfoutput>
    </cfif>

    <font face="Arial" size="-1">Type in your comment, then press "Save"<br>
15 <!-- Use LTime for current time --->
    <form action="AddComment.cfm?Edit=Yes" method="POST"
    onSubmit="setTimeout('window.close()',2000);">
    <cfoutput>
    <input type="Hidden" name="LTime" value="#LTime#">
20 <cfif ParameterExists(EventID)>
    <input type="Hidden" name="EventID" value="#EventID#">
    </cfif>
    <input type="Hidden" name="DS" value="#DS#">
    <input type="Hidden" name="Clipsrc" value="#Clipsrc#">
25 </cfoutput>
    <center>
    <textarea name="NewComment" cols="40" rows="2" wrap="VIRTUAL"></textarea>
    <input type="Submit" value="Save"></center>
    </font>
30 </form>
```

File: banner2.cfm

```
<html>
5  <head>
  <title></title>
  </head>
  <SCRIPT LANGUAGE = "JavaScript">
    <!-- hide this script from old browsers
10  LinkWindow="status=no,toolbar=yes,directories=no,menubar=yes,location=yes,scrollbars=y
    es,resizable=yes,HEIGHT=400,WIDTH=600";
    // -->
  </script>
  <body bgcolor="#5B94D1">
15    <CFIF ParameterExists(imgsrc) AND ParameterExists(msg2)>
      <cfoutput>
        <A href="http://#msg2#" target="NetWin"
onClicK="window.open('NetWin',LinkWindow);"><IMG align="left" alt="#caption#"
border=0 src="http://#imgsrc#"></a>
20      </cfoutput>
      <CFELSE>
        &nbsp;
      </CFIF>

25  </body>
  </html>
```

Prepared by: www.ck12.org

File: buttonbar2.cfm

```
<html>
5 <head>
  <title></title>
</head>

<SCRIPT LANGUAGE = "JavaScript">
10 name = 'mymainwindow';
  CaptionWin=null;
  <!-- hide this script from old browsers
  LinkWindow="status=no,toolbar=yes,directories=no,menubar=yes,location=no,scrollbars=yes,
  s,resizable=yes,HEIGHT=480,WIDTH=640";
15 SelectWindow="status=no,toolbar=yes,directories=no,menubar=yes,location=no,scrollbars=yes,
  es,resizable=yes,HEIGHT=400,WIDTH=600";
  function WinOpen()
  {
    if (CaptionWin == null)
20    {
      CaptionWin = window.open("caption2.cfm", "CaptionWin",
      "menubar=0,toolbar=0,resizable=0,width=320,height=360");
      CaptionWin.focus();
      document.forms[0].isWinOpen.value="true";
25      return false;
    }
    else
    {
      if (CaptionWin.closed == true)
30      {
        CaptionWin = window.open("caption2.cfm", "CaptionWin",
        "menubar=0,toolbar=0,resizable=0,width=320,height=360");
        CaptionWin.focus();
        document.forms[0].isWinOpen.value="true";
35        return false;
      }
    }
  }
  //function isWinOpen()
40  //{
  //  if (CaptionWin == null)
  //  {
  //    WinOpen();
  //    CaptionWin.hide();
45  //    return false;
  //  }
  }
```

```

//      else
//      {
//          if (CaptionWin.closed == true)
//          {
5 //              return false;
//          }
//          else
//          {
//              return true;
10 //          }
//      }
//}
// -->
</script>

15 <!-- buttonbar in frame name="ButtonWin" --->

<body bgcolor="#8FBC8F" text="Black" link="Blue" vlink="Blue" alink="#FF4500">
<!-- <cfif Channel is ">
20 onLoad="window.open('../Selector/index.cfm','SelectWin',LinkWindow);"
</cfif> --->
<table width='100%' border="0" cellspacing="0" cellpadding="0" bgcolor="#8FBC8F">
    <tr>
        <td width="10" rowspan="3">
25 background="../images/site/transparent.gif"></td>
        <td height="3" colspan="2" background="../images/site/transparent.gif"></td>
        <td rowspan="3" align="RIGHT" valign="TOP"></td>
30 </tr>
        <tr>
            <td><font face="Arial" size="-1">
                <a href="../Selector/channels.cfm" target="SelectWin"
onClick="window.open('','SelectWin',LinkWindow);">Channels</a>
35 </td><td>
                <font face="Arial" size="-1">
                    <a
href="../Links/index.cfm?ClipSrc=<cfoutput>#Clipsrc#</cfoutput>"
target="OutLinkWin" onClick="window.open('','OutLinkWin',LinkWindow);">Links</a>
40 </td>
        </tr>
        <tr><td>
            <font face="Arial" size="-1">

```

```

                                <font face="Arial" size="-1"> <a
href="../Selector/selections.cfm" target="SelectWin"
onClick="window.open('','SelectWin',LinkWindow);">Selections</a>
                                </td><td>
5                                <font face="Arial" size="-1">
                                <cfif NOT Channel is ">
                                <a
href="../chat/index.cfm?Channel=<cfoutput>#Channel#</cfoutput>" target="OutLinkWin"
onClick="window.open('','OutLinkWin',LinkWindow);">Chat</a>
10                                </cfif>
                                </td>
                                </tr>
                                <!--
                                <tr>
                                <td colspan="2">
15                                <font face="Arial" size="-1">
                                <cfif NOT Channel is ">
                                <a
href="caption2.cfm?Channel=<cfoutput>#Channel#</cfoutput>" target="CaptionWin"
onClick="return(WinOpen())">CC</a>
20                                </cfif>
                                </td>
                                </tr> --->
                                <tr><td height="3" colspan="4" background="../images/site/transparent.gif">
                                </td></tr>
25                                <tr><td colspan="4">
                                <table width="100%" border="0" cellspacing="0" cellpadding="2">
                                <form name='SearchForm' action="../Links/search.cfm" target="OutLinkWin"
onSubmit="window.open('','OutLinkWin',LinkWindow);
30 setTimeout('document.SearchForm.reset()',2000);" method="POST">
                                <input type="Hidden" name="isWinOpen" value="">
                                <tr bgcolor="#8FBC8F"><td align="CENTER">
                                <font face="Arial" size="1">
                                Find <input type="Text" name="SrchFor" size="15" maxlength="50"></font>
35                                <center><font size="1"><input type="Submit"
                                value="Search!"></font></center>
                                </td></tr>
                                </form>
                                </table>
40                                </td></tr>
                                </table>
                                </body>
                                </html>
45
```

File: caption2.cfm

```
<html>
<head>
5 <title>Closed Captioning</title>
  <script language="JavaScript">
    function SetUnLoad()
    {
10       opener.document.forms[0].isWinOpen.value="false";
        window.close();
    }
  </script>
</head>
<body bgcolor="ghostwhite">
15 <!-- sponsor in frame name="SponsorWin" --->
  <table width="100%" border="0" cellspacing="0" cellpadding="0" bgcolor="ghostwhite">
    <tr>
      <form name="cc">
        <td align='center'>
20           <table width="95%" border="0" cellspacing="0" cellpadding="0"><tr>
              <td>
                <CFIF ParameterExists(caption)>
                  <cfoutput>
                    <font size="-1">
25                     <textarea name='Caption' cols='40' rows='10'
                        wrap="VIRTUAL">#Replace(caption,"||","","ALL")#</textarea>
                    </font>
                  </cfoutput>
                </cfif>
30             </td>
          </tr>
        </table>
      </td>
    </TR>
35 </form>
  </TABLE>
</body>
</html>
```

File: comment2.cfm

```
<html>
5 <head>
  <title></title>
</head>
<SCRIPT LANGUAGE = "JavaScript">
<!-- hide this script from old browsers
10 CommWin=null;
LinkWindow="status=no,toolbar=yes,directories=no,menubar=yes,location=yes,scrollbars=y
es,resizable=yes,HEIGHT=400,WIDTH=600";
SmallWindow="status=no,toolbar=yes,directories=no,menubar=no,location=no,scrollbars=n
o,resizable=yes,HEIGHT=200,WIDTH=400";
15 function WinOpen()
{
    var ltime=parent.DataWin.document.Comments.ltime.value;
    var ltime_1=Math.floor(ltime/1000);
20    var clip=parent.DataWin.document.Comments.Clipsrc.value;
    var ds=parent.DataWin.document.Comments.DS.value;
    var What_url="addcomment.cfm?LTime="+ltime_1+"&ds="+ds+"&Clipsrc="+clip;
    if (CommWin == null)
    {
25        CommWin = window.open(What_url, "CommWin", SmallWindow);
        CommWin.focus();
    }
    else
    {
30        if (CommWin.closed == true)
        {
            CommWin = window.open(What_url, "CommWin", SmallWindow);
            CommWin.focus();
        }
35    }
}

function SetElapseTime()
40 {
    var ltime=parent.DataWin.document.Comments.ltime.value;
    var clip=parent.DataWin.document.Comments.Clipsrc.value;
    var ds=parent.DataWin.document.Comments.DS.value;
    location.href="addcomment.cfm?LTime="+ltime+"&ds="+ds+"&Clipsrc="+clip;
45 }
// -->
```







File: data2.cfm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
5  <html>
    <head>
        <title>Untitled</title>
    </head>
10  <body>
    <form name="Sponsor" action="sponsor2.cfm" method="post" enctype="application/x-
    www-form-urlencoded" target="SponsorWin">
        <input type="Hidden" name="imgsrc" value="">
        <input type="Hidden" name="msg2" value="">
15  <input type="Hidden" name="caption" value="">
        <input type="Hidden" name="Net" value="">
        <input type="Hidden" name="Channel" value="">
    </FORM>

20  <form name="Links" action="linklist2.cfm" method="post" enctype="application/x-www-
    form-urlencoded" target="LinkListWin">
        <input type="Hidden" name="msg" value="">
    </FORM>

25  <form name="Ad" action="banner2.cfm" method="post" enctype="application/x-www-form-
    urlencoded" target="BannerWin">
        <input type="Hidden" name="imgsrc" value="">
        <input type="Hidden" name="msg2" value="">
        <input type="Hidden" name="caption" value="">
30  </FORM>

    <form name="Caption" action="caption2.cfm" method="post" enctype="application/x-www-
    form-urlencoded" target="CaptionWin">
        <input type="Hidden" name="caption" value="">
35  </FORM>

    <form name="Comments" action="comment2.cfm" method="post" enctype="application/x-
    www-form-urlencoded" target="CommentWin">
        <input type="Hidden" name="ltime" value="0">
40  <input type="Hidden" name="msg" value="">
    <CFOUTPUT>
        <input type="Hidden" name="Clipsrc" value="#Clipsrc#">
        <input type="Hidden" name="DS" value="#ds#">
    </CFOUTPUT>
45  </FORM>
</body>
```

&lt;/html&gt;

File: linklist2.cfm

```
<html>  
5 <head>  
  <title></title>  
  <SCRIPT LANGUAGE = "JavaScript">  
    <!-- hide this script from old browsers  
    LinkWindow="status=no,toolbar=yes,directories=no,menubar=yes,location=yes,scrollbars=y  
10 es,resizable=yes,HEIGHT=400,WIDTH=600";  
    // -->  
  </script>  
  </head>  
  <body bgcolor="#F4A460" link="Blue" text="Black" vlink="#5B94D1" alink="#FF4500">  
15   <font face="Arial" size="1">  
     &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~  
     ~~~~~Current Links:  
     <br>  
     <b>  
       <CFIF ParameterExists(msg)>  
20       <cffoutput>  
         #Replace(msg,"||"," ",ALL)#  
       </cffoutput>  
       </cfif>  
     </font>  
25   </body>  
  </html>
```

File: main2.cfm

```
<cfparam name="Channel" default="">
5 <cfparam name="Net" default="">
  <cfparam name="Clipsrc" default="">
  <cfparam name="size" default="">
  <cfparam name="DS" default="Adserver">

10 <script language="JavaScript">
  <!--
  <!-- if (document.images) --->
    self.name = 'showtime';
  <!-->
15 </script>
  <!-- frames -->
  <frameset rows="*,0,62" framespacing="0" border="no" frameborder="0" marginwidth="0"
  marginheight="0" bordercolor="white">
    <frameset cols="169,*">
20       <frameset rows="132,90,*">
         <frame name="SponsorWin"
src="sponsor2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&ClipSrc=#Clipsrc#&DS=#
DS#</cfoutput>" framespacing="0" border="0" frameborder="0" marginwidth="0"
marginheight="0" bordercolor="white" scrolling="No" noresize>
25         <frame name="ButtonWin"
src="buttonbar2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size
=#Size#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0"
marginwidth="0" marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
         <frame name="LinkListWin"
30 src="linklist2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>" framespacing="0"
border="0" frameborder="0" marginwidth="0" marginheight="0" bordercolor="white"
scrolling="no" noresize frameborder="no">
         </frameset>
         <frame name="StreamWin"
35 src="stream2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size=#
Size#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0"
marginwidth="0" marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
         </frameset>
         <frame name="DataWin"
40 src="data2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size=#Siz
e#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0" marginwidth="0"
marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
         <frameset cols="*,468">
         <frame name="CommentWin"
45 src="comment2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>"
```

5

10

&lt;/frameset&gt;

File: main3.cfm

```
<cfparam name="Channel" default="">
5 <cfparam name="Net" default="">
  <cfparam name="Clipsrc" default="">
  <cfparam name="size" default="">
  <cfparam name="DS" default="Adserver">

10 <script language="JavaScript">
  <!--
  <!-- if (document.images) --->
    self.name = 'showtime';
  <!-->
15 </script>
  <!-- frames -->
  <frameset rows="*,62" framespacing="0" border="0" frameborder="0" marginwidth="0"
marginheight="0" bordercolor="white">
    <frameset cols="169,*,0">
20       <frameset rows="132,130,*">
         <frame name="SponsorWin"
src="sponsor2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&ClipSrc=#Clipsrc#&DS=#
DS#</cfoutput>" framespacing="0" border="0" frameborder="0" marginwidth="0"
marginheight="0" bordercolor="white" scrolling="No" noresize>
25         <frame name="ButtonWin"
src="buttonbar2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size
=#Size#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0"
marginwidth="0" marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
         <frame name="LinkListWin"
30 src="linklist2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>" framespacing="0"
border="0" frameborder="0" marginwidth="0" marginheight="0" bordercolor="white"
scrolling="no" noresize frameborder="no">
         </frameset>
         <frame name="StreamWin"
35 src="stream3.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size=#
Size#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0"
marginwidth="0" marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
         <frame name="DataWin"
src="data2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size=#Siz
40 e#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0" marginwidth="0"
marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
         </frameset>
         <frameset cols="*,468">
         <frame name="CommentWin"
45 src="comment2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>"
```





File: main4.cfm

```
5  <cfparam name="Channel" default="">
    <cfparam name="Net" default="">
    <cfparam name="Clipsrc" default="">
    <cfparam name="size" default="">
    <cfparam name="DS" default="Adserver">

10  <script language="JavaScript">
    <!--
    <!-- if (document.images) --->
        self.name = 'showtime';
    <!-->

15  </script>
    <!-- frames -->
    <frameset rows="*,62" framespacing="0" border="0" frameborder="0" marginwidth="0"
    marginheight="0" bordercolor="white">
        <frameset cols="169,*,0">

20            <frameset rows="132,130,*">
                <frame name="SponsorWin"
    src="sponsor2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&ClipSrc=#Clipsrc#&DS=#
    DS#</cfoutput>" framespacing="0" border="0" frameborder="0" marginwidth="0"
    marginheight="0" bordercolor="white" scrolling="No" noresize>

25                <frame name="ButtonWin"
    src="buttonbar2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size
    =#Size#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0"
    marginwidth="0" marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
                <frame name="LinkListWin"
    src="linklist2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>" framespacing="0"
    border="0" frameborder="0" marginwidth="0" marginheight="0" bordercolor="white"
    scrolling="no" noresize frameborder="no">
                </frameset>
            <frame name="StreamWin"
    src="stream4.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size=#
    Size#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0"
    marginwidth="0" marginheight="0" bordercolor="white" scrolling="no" frameborder="no">
            <frame name="DataWin"
    src="data2.cfm?Channel=<cfoutput>#Channel#&Net=#Net#&Clipsrc=#Clipsrc#&Size=#Siz
    e#&DS=#DS#</cfoutput>" framespacing="0" border="0" frameborder="0" marginwidth="0"
    marginheight="0" bordercolor="white" scrolling="no" frameborder="no">

40        </frameset>
        <frameset cols="*,468">
            <frame name="CommentWin"
    src="comment2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>"
```

```
framespacing="0" border="0" frameborder="0" marginwidth="0" marginheight="0"
bordercolor="white" scrolling="no" noresize frameborder="no">
    <frame name="BannerWin"
src="banner2.cfm?<cfoutput>Clipsrc=#Clipsrc#&DS=#DS#</cfoutput>" framespacing="0"
5 border="0" frameborder="0" marginwidth="0" marginheight="0" bordercolor="white"
scrolling="no" noresize frameborder="no">

    </frameset>
</frameset>
```

10

104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

File: search.cfm

```
<html>
5  <head>
  <title>Search</title>
  </head>
  <SCRIPT LANGUAGE="JavaScript">
    <!--
10   window.name="OutLinkWin";
    //-->
  </SCRIPT>

  <body onLoad="if (window.focus) self.focus(); else alert('Page Loaded');">
15

  search Window

20  <form>
    <input type="Button" value="Close" onClick="window.close();">
  </form>

25  </body>
  </html>
```

[illegible]

File: sponsor2.cfm

```

5  <cfquery name="GetNet" datasource="TV">
    Select* from NetStns
    Where Call = '#Net#'
  </cfquery>

  <cfquery name="GetChannel" datasource="TV">
10  Select* from NetStns
    Where Call = '#Channel#'
  </cfquery>

  <html>
15  <head>
    <title>sponsor</title>

    </head>
    <SCRIPT LANGUAGE = "JavaScript">
20  <!-- hide this script from old browsers
    LinkWindow="status=no,toolbar=yes,directories=no,menubar=yes,location=yes,scrollbars=y
    es,resizable=yes,HEIGHT=400,WIDTH=600";
    // -->
    function setSponsor(msg1)
25  {
    document.writeln(msg1);
    }
  </script>

30  <body bgcolor="#228B22">
    <!-- sponsor in frame name="SponsorWin" --->
    <table width="169" border="0" cellspacing="0" cellpadding="0" bgcolor="#228B22">
      <tr>
        <td width="30" rowspan="4" bgcolor="#8FBC8F">
35  <IMG align="middle" alt="" border=0 height=132 src="/images/site/sponsor_curve.gif"
            width=30></a>
        </TD>

        <td height="3" bgcolor="#228b22" background="transparent.gif">
40  </td>

      </TR>
      <tr>
        <td>
45  <table width="120" border="0" cellspacing="0" cellpadding="0">
      <tr>
```

```

    <td width="32" height="32">
    <cfif GetNet.recordcount gt 0>
    <cfoutput>
    <a href="http://#GetNet.LinkURL#" target="NetWin"
5  onClick="window.open('NetWin',LinkWindow);"><IMG align="middle"
alt="#GetNet.LinkText#" border=0 height=32 src="#getNet.ImageLoc#" width=32></a>
    </cfoutput>
    <cfelse>
    &nbsp;
10  </cfif>
    </td>

    <td>
    <a href="http://ilovetv.ca" target="new"><IMG align="middle" alt="" border=0
15  height=32 src="/images/ads/network/love.jpg" width=32></a>
    </td>
    <td width="32" height="32">
    <cfif GetChannel.recordcount GT 0>
    <cfoutput>
    <a href="http://#GetChannel.LinkURL#" target="NetWin"
20  onClick="window.open('NetWin',LinkWindow);"><IMG align="middle"
alt="#GetChannel.LinkText#" border=0 height=32 src="#getChannel.ImageLoc#"
width=32></a>
    </cfoutput>
    <cfelse>
    &nbsp;
25  </cfif>
    </td>
    </tr>
30  </table>
</td>
</TR>
<tr>
<td width="120" height="90">
35  <CFIF ParameterExists(imgsrc) AND ParameterExists(msg2)>
    <cfoutput>
    <A href="http://#msg2#" target="NetWin"
onClick="window.open('NetWin',LinkWindow);"><IMG align="left" alt="#caption#"
border=0 src="http://#imgsrc#"></a>
40  </cfoutput>
    <CFELSE>
    &nbsp;
    </CFIF>
    </TD>
45  </tr>
</tr>
```

&lt;/TABLE&gt;

&lt;/html&gt;&gt;



File: stream2.cfm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
5  <html>
  <head>
    <title>Untitled</title>
  </head>
10  <cfif Trim(Channel) is "or channel is ' ">
    <body bgcolor="#228B22" leftmargin=3 topmargin=3>
      <!--
        <a href="http://home.sol.no/~ggunners/NetBird.html">Net Bird</a>
15      --->
        <br><br><br>
        <table width="80%" align="CENTER" valign="MIDDLE" bgcolor="#F4A460">
          <tr><td>
            <font face="Arial" size="+1" color="Blue">Select a program to view.<br><br>
20            Selections button presents archived shows for this demo.<br><br>
            Channels button presents a channel selector mock-up.<br><br>
            Links button shows all the links from the current program.<br><br>
            Almost everything on the screen is a live link to the internet.
          </td>
25  </tr>
        </table>
        &nbsp; <br>

30      </center>
      </font>

    </body>
  </html>
35  <cfelse>

    <script language="javascript">
40  <!--
    var count = 0;
    var doPolling;
    var Ver4 = parseInt(navigator.appVersion) >= 4;
    var Nav4 = ((navigator.appName == "Netscape") && Ver4);
45  var IE4 = ((navigator.userAgent.indexOf("MSIE") != -1) && Ver4);
```

```
var Messages = "";
var msg="";
var Current_time=0;
//caption
5 var c_msg=";
var c_count=0;
//link
var l_count=0;
var l_msg=";
10 //comment
var m_count=0;
var m_msg=";

var isAppletReady=false;
15 CaptionWin=null;
<CFOUTPUT>
var Net='#Net#';
var Channel='#Channel#';
</CFOUTPUT>
20 // Save window content dimensions to work around resize
// behaviour in WIN32 versions of Navigator 4.
if (Nav4)
{
    var origWidth = window.innerWidth;
    var origheight = window.innerHeight;
25 }

function WinOpen()
{
30     if (CaptionWin == null)
    {
        CaptionWin = window.open("caption2.cfm", "CaptionWin",
"menubar=0,toolbar=0,resizeable=0,width=320,height=180");
        CaptionWin.focus();
35     return false;
    }
    else
    {
        if (CaptionWin.closed == true)
40     {
            CaptionWin = window.open("caption2.cfm", "CaptionWin",
"menubar=0,toolbar=0,resizeable=0,width=320,height=360");
            CaptionWin.focus();
            return false;
45     }
    }
}
```

```
}
// Invoked by <body> tag's onResize event handler to refresh the page for
// Navigator 4 to assist in getting positioned elements back in place.
function RestorePage()
5  {
    if (Nav4)
    {
        if (origWidth != window.innerWidth || origHeight != window.innerHeight)
10         history.go(0);
    }
}

function SendChatMessage(Author, Type, Message)
15 {
    document.applets[0].SendMessage(Author, Type, Message);
}

function SendElapsedTimeToApplet(Time)
20 {
    //var tx=Time/1000;
    //var tx1=Math.floor(tx);
    document.applets[0].SendElapsedTime(Time);
}
25

function HandleSessionMessageEx(Author, Type, Message)
{
    if (IE4)
30     {
        document.forms[0].WhatPosition.value=Message;
    }
    else if (Nav4)
    {
35     document.forms[0].WhatPosition.value=Message;
    }
}

40
function HandleAppletLoad(Message)
{
    isAppletReady=true;
}

45
function HandleElapsedTime(Message)
```

```
{
    if (IE4)
    {
        document.forms[0].WhatPosition.value=Message;
5      }
      else if (Nav4)
      {
        document.forms[0].WhatPosition.value=Message;
10     }
    }
    function HandleSponsor(Message1,Message2,Message3)
    {
15      //Handle the images in the SponsorWin frame
      parent.DataWin.document.Sponsor.imgsrc.value = Message1;
      parent.DataWin.document.Sponsor.msg2.value = Message2;
      parent.DataWin.document.Sponsor.caption.value = Message3;
      parent.DataWin.document.Sponsor.Net.value = Net;
20     parent.DataWin.document.Sponsor.Channel.value = Channel;

      HandleLinks(Message2,Message3,"");

      parent.DataWin.document.Sponsor.submit();
25  }
    function HandleLinks(Message1,Message2, Message3)
    {
      var Type=Message3;
      if(Type == "old")
30     {
        l_msg="<a href='http://'+Message1+'\" target='NetWin'
onClick=\"window.open(\"',NetWin',LinkWindow)\\\">"+Message2+"</a><br>||"+l_msg;
      }
      else
35     {
        l_msg="<a href='http://'+Message1+'\" target='NetWin'
onClick=\"window.open(\"',NetWin',LinkWindow)\\\">"+Message2+"</a><br>||"+l_msg;
      }
40     l_count++;

      if(l_count >5)
      {
        var last=l_msg.lastIndexOf("||");
45     l_msg = l_msg.substring(0,last);
      }
    }
  }
```

```
parent.DataWin.document.Links.msg.value = l_msg;
parent.DataWin.document.Links.submit();

5
}
function HandleAd(Message1,Message2, Message3)
{
    //Handle the images in the SponsorWin frame
10    parent.DataWin.document.Ad.imgsrc.value = Message1;
    parent.DataWin.document.Ad.msg2.value = Message2;
    parent.DataWin.document.Ad.caption.value = Message3;
    HandleLinks(Message2,Message3, "");
    parent.DataWin.document.Ad.submit();
15 }
function HandleCaption(msg)
{
    if(msg != "")
    {
20        if(c_count > 5)
        {
            var last=c_msg.lastIndexOf("||");
            c_msg = c_msg.substring(0,last);
            c_msg=msg+"\r\n||"+c_msg;
25        }
        else
        {
            c_msg=msg+"\r\n||"+c_msg;
        }
30        //Handle the images in the SponsorWin frame
        if(CaptionWin == null || CaptionWin.closed == true)
        {
            }
            else
35            {
                parent.DataWin.document.Caption.caption.value = c_msg;
                parent.DataWin.document.Caption.submit();
            }
40        c_count++;
    }
}

45 function HandleComments(msg)
{
```

```

    if(msg !=")
    {
        if(m_count >5)
        {
            5          var last=m_msg.lastIndexOf("||");
                      m_msg = m_msg.substring(0,last);
                      m_msg=msg+"<BR>||"+m_msg;
        }
        else
        10         {
                    m_msg=msg+"<BR>||"+m_msg;
                }
        parent.DataWin.document.Comments.msg.value = m_msg;
        parent.DataWin.document.Comments.submit();
        15         m_count++;
    }
}

20 function HandleSessionErrorEx(ErrNumber, ErrDescription)
{
    alert(ErrDescription);
}

25 function SendElapsedTime()
{
    if(document.video1.GetPlayState()== 3)
    30     {
        // Make sure that the applet has been loaded
        if(isAppletReady)
        {
            35             var ElpTime=document.video1.GetPosition();
                        if(ElpTime != "")
                        {
                            SendElapsedTimeToApplet(ElpTime);
                            parent.DataWin.document.Comments.ltime.value = ElpTime;
                        }
            40             polling();
        }
        else
        {
            45             alert("The Applet on your browser has not been started. Please refresh
                        your page");
                        stopPolling();
                    }
                }
            }
        }
    }
}

```

```
        }
    }
    else
    {
5      doPolling = setTimeout("SendElapsedTime()",500);
    }
}

function stopPolling()
10 {
    clearTimeout(doPolling)
}
function polling()
{
15    doPolling = setTimeout("SendElapsedTime()",1000);
}

//-->
20 </script>

<body bgcolor="#228B22" leftmargin=3 topmargin=3>

    <applet id="ChatApplet" width="1" height="1" code="ChatClient.class"
25    codebase="http://blanche.tcn.net:8090/" VIEWASTEXT mayscript>
    <param name="MessageHandler" value="HandleSessionMessageEx">
    <param name="ElapsedTimeMessageHandler" value="HandleElapsedTime">
    <param name="ErrorHandler" value="HandleSessionErrorEx">
    <param name="SponsorHandler" value="HandleSponsor">
30    <param name="LinksHandler" value="HandleLinks">
    <param name="CaptionHandler" value="HandleCaption">
    <param name="AdHandler" value="HandleAd">
    <param name="CommentsHandler" value="HandleComments">
    <param name="CaptionHandler" value="HandleCaption">
35    <param name="ErrorHandler" value="HandleSessionErrorEx">
    <param name="TimeHandler" value="HandleTime">
    <param name="isAppletLoad" value="HandleAppletLoad">

    <CFOUTPUT>
40    <param name="ClipSource" value="#ClipSrc#">
    </CFOUTPUT>
    </applet>
    <script Language="JavaScript">
        polling();
45    </script>
```

```
<table width="100%" border="0" cellspacing="0" cellpadding="0" align="LEFT"
valign="TOP">
<tr>
<td align="left" valign="top">
5      <!-- stream in frame name="streamWin" --->

      <cfparam name="Channel" default="">
      <cfparam name="Net" default="">
      <cfparam name="Clipsrc" default="">
10     <cfparam name="size" default="">

      <cfif Size is 'S'>
          <cfset Width=320>
          <cfset Height=240>
15     <cfelseif Size is 'L'>
          <cfset Width=512>
          <cfset Height=384>
          <cfelseif Size is 'X'>
          <cfset Width=240>
20     <cfset Height=180>
      </cfif>

      <cfif Not Clipsrc is ">
          <cfif ClipSrc is "SunPar3" and Size is "L">
25     <cfset cliploc =
'http://www.broadcast.com/business/sun/98par3challenge_102099_100.ram'>
          <cfelseif ClipSrc is "SunPar3" and Size is "S">
          <cfset cliploc =
'http://www.broadcast.com/business/sun/98par3challenge_102099_56.ram'>
30     <cfelseif ClipSrc is "Live001" and Size is "S">
          <cfset cliploc = '/media/56k/Live001.ram'>

          <cfelseif ClipSrc is "Auction" and Size is "S">
35     <cfset cliploc = '/media/56k/auction.ram'>
          <cfelse>
          <cfif Size is 'S'>
              <cfset cliploc = '/media/56k/#Clipsrc#.ram'>
              <cfelseif Size is 'L'>
40     <cfset cliploc = '/media/108k/#Clipsrc#.ram'>
          <cfelse>
              <cfset cliploc = '/media/#Clipsrc#.ram'>
          </cfif>
          </cfif>
45     </cfif>
</cfif>
```



```
<center>
<cfoutput>
<object
5  ID=video1
   CLASSID="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA"
   height=#Height#
   width=#Width#
   >
10  <param name="controls" value="ImageWindow">
    <param NAME="console" VALUE="Clip1">
    <param NAME="autostart" VALUE="true">
    <param NAME="src" VALUE="#Cliploc#">
15  <embed name="video1" SRC="#Cliploc#"
    type="audio/x-pn-realaudio-plugin"
    CONSOLE="Clip1"
    controls="ImageWindow"
    height=#Height#
20  width=#Width#
    AUTOSTART=true>
    </object>

    <BR>
25  <form>
    <input type="Button" name="Play_Button" value="Play"
onClick="document.applets[0].init();document.video1.DoPlay();polling()">
    <input type="Button" name="Pause_Button" value="Pause"
onClick="document.video1.DoPause();stopPolling();">
30  <input type="Button" name="Stop_Button" value="Stop"
onClick="document.video1.DoStop();stopPolling();">
    <!-- <input type="Button" name="position" value="Position"
onClick="javascript:alert(document.video1.GetPosition())">
    <input type="Button" name="buffer" value="Send"
35  onClick="SendMessage('ripu', 'ChatMessage', document.video1.GetPosition())">
    <input type="Text" name="WhatPosition"> --->
    </form>
    </cfoutput>
    </center>
40 </td>
    </tr>
    <tr>
    <td colspan="2">
        <font face="Arial" size="-1">
45  <a
href="caption2.cfm?Channel=<cfoutput>#Channel#</cfoutput>" target="CaptionWin"
```

5

5

File: stream3.cfm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

5  <html>
    <head>
        <title>Untitled</title>
    </head>

10  <cfif Trim(Channel) is "or channel is ' '>
    <body bgcolor="#228B22" leftmargin=3 topmargin=3>

        <a href="http://home.sol.no/~ggunners/NetBird.html">Net Bird</a>

15      <br><br><br>
        <table width="80%" align="CENTER" valign="MIDDLE" bgcolor="#F4A460">
            <tr><td>
                <font face="Arial" size="+1" color="Blue">Select a program to view.<br><br>
                Selections button presents archived shows for this demo.<br><br>
20      Channels button presents a channel selector mock-up.<br><br>
                Links button shows all the links from the current program.<br><br>
                Almost everything on the screen is a live link to the internet.
            </td>
        </tr>

25      </table>
        &nbsp; <br>

        </center>
30      </font>

    </body>
    </html>
    <cfelse>

35      <script language="javascript">
        <!--
        var count = 0;
40      var doPolling;
        var Ver4 = parseInt(navigator.appVersion) >= 4;
        var Nav4 = ((navigator.appName == "Netscape") && Ver4);
        var IE4 = ((navigator.userAgent.indexOf("MSIE") != -1) && Ver4);

45      var Messages = "";
        var msg="";
```

```
var c_msg="";
var Current_time=-1000;
CaptionWin=null;
var c_count=0;
5 // Save window content dimensions to work around resize
  // behaviour in WIN32 versions of Navigator 4.
  if (Nav4)
  {
    var origWidth = window.innerWidth;
10    var origheight = window.innerHeight;
  }

  // Invoked by <body> tag's onResize event handler to refresh the page for
15 // Navigator 4 to assist in getting positioned elements back in place.
  function RestorePage()
  {
    if (Nav4)
    {
20      if (origWidth != window.innerWidth || origHeight != window.innerHeight)
        history.go(0);
    }
  }

25 function SendChatMessage(Author, Type, Message)
  {
    document.applets[0].SendMessage(Author, Type, Message);
  }

30 function SendElapsedTimeToApplet(Time)
  {
    document.applets[0].SendElapsedTime(Time);
  }

35 function HandleSessionMessageEx(Author, Type, Message)
  {
    if (IE4)
    {
40      document.forms[0].WhatPosition.value=Message;
    }
    else if (Nav4)
    {
45      document.forms[0].WhatPosition.value=Message;
```

```
    }  
}  
  
5  function HandleElapsedTime(Message)  
    {  
        if (IE4)  
        {  
            document.forms[0].WhatPosition.value=Message;  
10     }  
        else if (Nav4)  
        {  
            document.forms[0].WhatPosition.value=Message;  
15     }  
    }  
}  
function HandleSponsor(Message1,Message2,Message3)  
{  
20     //Handle the images in the SponsorWin frame  
    parent.DataWin.document.Sponsor.imgsrc.value = Message1;  
    parent.DataWin.document.Sponsor.msg2.value = Message2;  
    parent.DataWin.document.Sponsor.caption.value = Message2;  
    parent.DataWin.document.Sponsor.submit();  
25 }  
function HandleLinks(Message1,Message2, Message3)  
{  
    var Type=Message3;  
30     if(Type == "old")  
    {  
        msg=msg+"<a href='"+Message1+"' target='NetWin'  
onClick=window.open('','NetWin',LinkWindow);>"+Message2+"</a><br>";  
    }  
35     else  
    {  
        msg="<a href='"+Message1+"' target='NetWin'  
onClick=window.open('','NetWin',LinkWindow);>"+Message2+"</a><br>";  
    }  
40     parent.DataWin.document.Links.msg.value = msg;  
    parent.DataWin.document.Links.submit();  
  
45 }  
function HandleTime(Message)
```

```
{
    Current_time=Message;
}

5 function HandleAd(Message1,Message2, Message3)
{
    //Handle the images in the SponsorWin frame
    parent.DataWin.document.Ad.imgsrc.value = Message1;
    parent.DataWin.document.Ad.msg2.value = Message2;
10    parent.DataWin.document.Ad.caption.value = Message2;
    parent.DataWin.document.Ad.submit();
}
function WinOpen()
{
15    if (CaptionWin == null)
    {
        CaptionWin = window.open("caption2.cfm", "CaptionWin",
"menubar=0,toolbar=0,resizeable=0,width=320,height=360");
        CaptionWin.focus();
20        return false;
    }
    else
    {
        if (CaptionWin.closed == true)
25        {
            CaptionWin = window.open("caption2.cfm", "CaptionWin",
"menubar=0,toolbar=0,resizeable=0,width=320,height=360");
            CaptionWin.focus();
            return false;
30        }
    }
}

function HandleCaption(msg)
35 {
    if(msg != "")
    {
        if(c_count > 5)
        {
40            var first=c_msg.indexOf("||");
            c_msg = c_msg.substring(first+2);
            c_msg=c_msg+"\r\n||"+msg;
        }
        else
45        {
            c_msg=c_msg+"\r\n||"+msg;
        }
    }
}
```

```
    }
    //Handle the images in the SponsorWin frame
    if(CaptionWin == null || CaptionWin.closed == true)
    {
5        }
        else
        {
            parent.DataWin.document.Caption.caption.value = c_msg;
10            parent.DataWin.document.Caption.submit();
        }
        c_count++;
    }
15 }

function HandleComments(Message)
{
    if (IE4)
    {
20        document.forms[0].WhatPosition.value=Message;
    }
    else if (Nav4)
    {
25        document.forms[0].WhatPosition.value=Message;
    }
}

30 function HandleSessionErrorEx(ErrNumber, ErrDescription)
{
    alert(ErrDescription);
}
35

function SendElapsedTime()
{
40        SendElapsedTimeToApplet(Current_time);
        Current_time+=1000;
        polling();
}

45 function stopPolling()
{
```

```
        clearTimeout(doPolling)
    }
    function polling()
    {
5        doPolling = setTimeout("SendElapsedTime()",1000);
    }
    //-->
</script>
10 <CFOUTPUT>
    <body bgcolor="##228B22" leftmargin=3 topmargin=3>
</CFOUTPUT>
15 <applet id="ChatApplet" width="1" height="1" code="ChatClient.class"
    codebase="http://blanche.tcn.net:8090/" VIEWASTEXT mayscript>
    <param name="MessageHandler" value="HandleSessionMessageEx">
    <param name="ElapsedTimeMessageHandler" value="HandleElapsedTime">
    <param name="ErrorHandler" value="HandleSessionErrorEx">
20 <param name="SponsorHandler" value="HandleSponsor">
    <param name="LinksHandler" value="HandleLinks">
    <param name="CaptionHandler" value="HandleCaption">
    <param name="AdHandler" value="HandleAd">
    <param name="CommentsHandler" value="HandleComments">
25 <param name="CaptionHandler" value="HandleCaption">
    <param name="ErrorHandler" value="HandleSessionErrorEx">
    <param name="TimeHandler" value="HandleTime">
    <CFOUTPUT>
    <param name="ClipSource" value="#ClipSrc#">
30 </CFOUTPUT>
    </applet>
    <script Language="JavaScript">
        polling();
        document.applets[0].setStartTime();
35 </script>
    <table width="100%" border="0" cellpadding="0" cellspacing="0" align="LEFT"
        valign="TOP">
        <tr>
        <td align="left" valign="top" width="50%">
40 <BR>
        </center>
        </td>
        </tr>
        <tr>
45 <td colspan="2">
            <font face="Arial" size="-1">
```





File: stream4.cfm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
5  <html>
    <head>
        <title>Untitled</title>
    </head>
10  <cfif Trim(Channel) is "or channel is ' '>
    <body bgcolor="#228B22" leftmargin=3 topmargin=3>

        <a href="http://home.sol.no/~ggunners/NetBird.html">Net Bird</a>
15      <br><br><br>
        <table width="80%" align="CENTER" valign="MIDDLE" bgcolor="#F4A460">
            <tr><td>
                <font face="Arial" size="+1" color="Blue">Select a program to view.<br><br>
20      Selections button presents archived shows for this demo.<br><br>
                Channels button presents a channel selector mock-up.<br><br>
                Links button shows all the links from the current program.<br><br>
                Almost everything on the screen is a live link to the internet.
            </td>
25      </tr>
        </table>
        &nbsp; <br>

30      </center>
        </font>

        </body>
        </html>
35  <cfelse>

        <script language="javascript">
            <!--
40      var count = 0;
            var doPolling;
            var Ver4 = parseInt(navigator.appVersion) >= 4;
            var Nav4 = ((navigator.appName == "Netscape") && Ver4);
            var IE4 = ((navigator.userAgent.indexOf("MSIE") != -1) && Ver4);
45      var Messages = ";
```

```
var msg="";
var c_msg="";
var Current_time=-1000;
CaptionWin=null;
5 // Save window content dimensions to work around resize
  // behaviour in WIN32 versions of Navigator 4.
  if (Nav4)
  {
    var origWidth = window.innerWidth;
10    var origheight = window.innerHeight;
  }

  // Invoked by <body> tag's onResize event handler to refresh the page for
15 // Navigator 4 to assist in getting positioned elements back in place.
  function RestorePage()
  {
    if (Nav4)
    {
20      if (origWidth != window.innerWidth || origHeight != window.innerHeight)
        history.go(0);
    }
  }

25 function SendChatMessage(Author, Type, Message)
  {
    document.applets[0].SendMessage(Author, Type, Message);
  }

30 function SendElapsedTimeToApplet(Time)
  {
    document.applets[0].SendElapsedTime(Time);
  }

35 function HandleSessionMessageEx(Author, Type, Message)
  {
    if (IE4)
40    {
      document.forms[0].WhatPosition.value=Message;
    }
    else if (Nav4)
    {
45    document.forms[0].WhatPosition.value=Message;
```

```
    }  
}  
  
5  function HandleElapsedTime(Message)  
    {  
        if (IE4)  
        {  
            document.forms[0].WhatPosition.value=Message;  
10     }  
        else if (Nav4)  
        {  
            document.forms[0].WhatPosition.value=Message;  
15     }  
    }  
}  
function HandleSponsor(Message1,Message2,Message3)  
10 {  
    //Handle the images in the SponsorWin frame  
    parent.DataWin.document.Sponsor.imgsrc.value = Message1;  
    parent.DataWin.document.Sponsor.msg2.value = Message2;  
    parent.DataWin.document.Sponsor.caption.value = Message2;  
    parent.DataWin.document.Sponsor.submit();  
25 }  
function HandleLinks(Message1,Message2, Message3)  
    {  
        var Type=Message3;  
30     if(Type == "old")  
        {  
            msg=msg+"<a href='"+Message1+"' target='NetWin'  
onClick=window.open('','NetWin',LinkWindow);>" +Message2+"</a><br>";  
        }  
35     else  
        {  
            msg="<a href='"+Message1+"' target='NetWin'  
onClick=window.open('','NetWin',LinkWindow);>" +Message2+"</a><br>";  
40     }  
    parent.DataWin.document.Links.msg.value = msg;  
    parent.DataWin.document.Links.submit();  
45 }  
function HandleTime(Message)
```

```
{
    Current_time=Message;
}

5 function HandleAd(Message1,Message2, Message3)
{
    //Handle the images in the SponsorWin frame
    parent.DataWin.document.Ad.imgsrc.value = Message1;
    parent.DataWin.document.Ad.msg2.value = Message2;
10    parent.DataWin.document.Ad.caption.value = Message2;
    parent.DataWin.document.Ad.submit();
}
function WinOpen()
{
15    if (CaptionWin == null)
    {
        CaptionWin = window.open("caption2.cfm", "CaptionWin",
        "menubar=0,toolbar=0,resizeable=0,width=320,height=360");
        CaptionWin.focus();
20        return false;
    }
    else
    {
        if (CaptionWin.closed == true)
25        {
            CaptionWin = window.open("caption2.cfm", "CaptionWin",
            "menubar=0,toolbar=0,resizeable=0,width=320,height=360");
            CaptionWin.focus();
            return false;
30        }
    }
}

function HandleCaption(msg)
35 {
    c_msg=c_msg+"\r\n"+msg;
    //Handle the images in the SponsorWin frame
    if(CaptionWin == null || CaptionWin.closed == true)
    {
40    }
    else
    {
        parent.DataWin.document.Caption.caption.value = c_msg;
45        parent.DataWin.document.Caption.submit();
    }
}
```

```
    }

    function HandleComments(Message)
    {
5      if (IE4)
        {
          document.forms[0].WhatPosition.value=Message;
        }
        else if (Nav4)
10       {
          document.forms[0].WhatPosition.value=Message;
        }
    }

15 }

    function HandleSessionErrorEx(ErrNumber, ErrDescription)
    {
20       alert(ErrDescription);
    }

    function SendElapsedTime()
25     {

        SendElapsedTimeToApplet(Current_time);
        Current_time+=1000;
        polling();
30     }

    function stopPolling()
    {
        clearTimeout(doPolling)
35     }
    function polling()
    {

        doPolling = setTimeout("SendElapsedTime()",1000);
40     }
    //-->
    </script>

    <CFOUTPUT>
45   <body bgcolor="##228B22" leftmargin=3 topmargin=3>
    </CFOUTPUT>
```

```
<applet id="ChatApplet" width="1" height="1" code="LiveChatClient.class"
codebase="http://blanche.tcn.net:8090/" VIEWASTEXT mayscript>
<param name="MessageHandler" value="HandleSessionMessageEx">
5 <param name="ElapsedTimeMessageHandler" value="HandleElapsedTime">
  <param name="ErrorHandler" value="HandleSessionErrorEx">
  <param name="SponsorHandler" value="HandleSponsor">
  <param name="LinksHandler" value="HandleLinks">
  <param name="CaptionHandler" value="HandleCaption">
10 <param name="AdHandler" value="HandleAd">
  <param name="CommentsHandler" value="HandleComments">
  <param name="CaptionHandler" value="HandleCaption">
  <param name="ErrorHandler" value="HandleSessionErrorEx">
  <param name="TimeHandler" value="HandleTime">
15 <CFOUTPUT>
  <param name="ClipSource" value="#ClipSrc#">
</CFOUTPUT>
</applet>
<script Language="JavaScript">
20 polling();
  document.applets[0].getCurrentPlayTime();
</script>
<table width="100%" border="0" cellspacing="0" cellpadding="0" align="LEFT"
valign="TOP">
25 <tr>
  <td align="left" valign="top">
    <!-- stream in frame name="streamWin" --->

    <cfparam name="Channel" default="">
30 <cfparam name="Net" default="">
    <cfparam name="Clipsrc" default="">
    <cfparam name="size" default="">

    <cfif Size is 'S'>
35 <cfset Width=160>
    <cfset Height=120>
    <cfelseif Size is 'L'>
    <cfset Width=512>
    <cfset Height=384>
40 </cfif>

    <cfif Not Clipsrc is ">
    <cfif ClipSrc is "SunPar3" and Size is "L">
    <cfset cliploc =
45 'http://www.broadcast.com/business/sun/98par3challenge_102099_100.ram'>
    <cfelseif ClipSrc is "SunPar3" and Size is "S">
```

```

        <cfset cliploc =
'http://www.broadcast.com/business/sun/98par3challenge_102099_56.ram'>
        <cfelse>
            <cfif Size is 'S'>
5              <cfset cliploc = '/media/56k/#Clipsrc#.ram'>
            <cfelse>
                <cfset cliploc = '/media/#Clipsrc#.ram'>
            </cfif>
        </cfif>
10    </cfif>

    <center>
    <cfoutput>
15    <object
        ID=video1
        CLASSID="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA"
        height=#Height#
        width=#Width#
20    >

        <param name="controls" value="ImageWindow">
        <param NAME="console" VALUE="Clip1">
        <param NAME="autostart" VALUE="true">
25    <param NAME="src" VALUE="#Cliploc#"
        <embed name="video1" SRC="#Cliploc#"
            type="audio/x-pn-realaudio-plugin"
            CONSOLE="Clip1"
            controls="ImageWindow"
30    height=#Height#
            width=#Width#
            AUTOSTART=true>
        </object>

35    <BR>
        <form>
            <input type="Button" name="Play_Button" value="Play"
onClick="document.applets[0].init();document.applets[0].getCurrentPlayTime();document.vi
deo1.DoPlay();polling()">
40    <input type="Button" name="Pause_Button" value="Pause"
onClick="document.video1.DoPause();stopPolling();">
            <input type="Button" name="Stop_Button" value="Stop"
onClick="document.video1.DoStop();stopPolling()">
            <!-- <input type="Button" name="position" value="Position"
45    onClick="javascript:alert(document.video1.GetPosition())">

```

For more see: www.90000.com



```

        <input type="Button" name="buffer" value="Send"
onClick="SendMessage('ripu', 'ChatMessage', document.video1.GetPosition())">
        <input type="Text" name="WhatPosition"> --->
        </form>
5      </cfoutput>
      </center>
    </td>
  </tr>
  <tr>
10  <td colspan="2">
        <font face="Arial" size="-1">
          <a
href="caption2.cfm?Channel=<cfoutput>#Channel#</cfoutput>" target="CaptionWin"
onClick="return(WinOpen())">CC</a>
15  </td>
    </tr>
  </table>
</body>
</html>
20 </cfif>

<!-- <table width="100%" border="0" cellspacing="0" cellpadding="0" align="LEFT"
valign="TOP">
25 <tr>
  <td align="left" valign="top" width="50%">
    <BR>
    </center>
  </td>
30 </tr>
  <tr>
    <td colspan="2">
        <font face="Arial" size="-1">
          <a
35 href="caption2.cfm?Channel=<cfoutput>#Channel#</cfoutput>" target="CaptionWin"
onClick="return(WinOpen())">CC</a>
    </td>
  </tr>
</table>
40 </body>
</html>
</cfif> --->
```

File: thankyou.cfm

```
5      <cfquery name="getcomments"
        datasource="VBICContent"
        username="caption"
        password="trewq55">

10      select comment from ctfiler
        where clipsource = '#clipsrc#'
        and Type = 'C'
        and seconds > #Evaluate(LTime -10)#
        and seconds < #Evaluate(LTime +10)#
        order by seconds

15      </cfquery>

20      <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

        <html>
        <head>
            <title>Thank You</title>
        </head>

25        <body>
            <font face="Arial" color = "cadetblue" size="-1">Thank you for your comment. Here are the
            comments around yours:
            <font color="crimson">
30            <br>
            <cfoutput query="GetComments">
                #comment#<br>
            </cfoutput></font>
            </font>

35        <form>
            <input type="Button" value="Close" onClick="window.close();">
        </form>
        </body>
        </html>

40
```

## Appendix D

Copyright © 2000 I LOVE TV INC. (868970 Ontario Inc.)  
All rights reserved unless otherwise noted

5

### Cold Fusion

### TESsocket

10

File: captionsock.c

15

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <fcntl.h>
#include <errno.h>
```

20

```
#define EOCAP (buf[0]==0x01 || buf[0]==0x02 || buf[0]==0x11 || buf[0]==0x12)
#define LOG(X) {fprintf(fp,"n%s",X); fflush(fp);}
#define DEBUG 0
```

25

```
#define captionsock 9000
#define IPTES3 "206.221.245.75"
```

```
FILE *fp;
```

30

```
#define stderr fp
```

```
static parse();
```

35

```
int argc;
char **argv;
int state;
char outstr[10000];
int sock_desc; /* Socket descriptor */
struct sockaddr_in sa; /* Internet Socket addr. struct */
40 struct hostent *hp; /* Result of host name lookup */
struct servent *sp; /* Result of service lookup */
int len,seq,i,flags=0;
char seq_str[30];
char rep_type;
45 char query_str[90000];
FILE *fp_head,*fp_list,*fp_tail;
```

```
char ch;
char ARGV[26][9000];
int ARGV;

5  int bargc;
    char *bargv[20],*cp;

    char myname[] = "captionsock";          /* this program name */

10  int inc;
    char junk[9000];
    char buf[9000];

15  /* all remaining test passed to program is then passed
    directly to upstream */

    int count;
    int lenr,res;
20  char blank[]="";

    #define MAXHOSTNAME 32
    #define BUFSIZE 9000
25  int
    main ( )
    {

    char sendbuff[300]; /* due to captioning data size limitations,
30                          this should be more than enough */

    int buffin;
    char *buffpnt;
    int firsttime=1;
    char field[2];
35  char outb[1000];
    char systemstring[200];

    char localhost[32];

40  fp=fopen("/tmp/captionlog","w");
    /* caption log stores raw data!!! */

    /******
    /******
45  gethostname(localhost, MAXHOSTNAME);
```

```
if((hp = gethostbyname(localhost)) == NULL) {
    fprintf(stderr,"no such host?\n");
    exit(0);
5  }

    sa.sin_port = htons(captionsock);
    bcopy((char *)hp->h_addr, (char *)&sa.sin_addr, hp->h_length);
    sa.sin_family = hp->h_addrtype;
10  inet_aton(IPTES3,&sa.sin_addr);
    /* IP of TES3!! */

    if((sock_desc=socket(hp->h_addrtype,SOCK_STREAM,0)) < 0) {
        perror("socket");
        exit(0);
15  }

    LOG("CONNECT SOCKET");
20  if(connect(sock_desc,&sa,(int)(sizeof sa)) < 0) {
        perror("connect");
        exit(0);
    }

25  LOG("READ");
    /* send these lines off to mysql as they come in */
    /* insert into record incrementing table, with time stamp on creation */
    /* perl program will collect together fragmented time stamped data,
       and join to completed objects ... then on from there*/
30  /* TOKENS at this time are 0x01 0x02 0x11 0x12 0x0a */

    field[1]='\0';

35  buffpnt=sendbuff;

    /* on first in, discard everything up to first valid TOKEN */

    for (;;) {
40  len = read(sock_desc,buf,1);
        buf[len]='\0';
        buf[0]=buf[0]&0x7f; /* mask of high bit */
        /* DEBUG: printf("%s",buf); */

45  if( len > 0 )
        {
```

```
if( EOCAP )
{
    *buffpnt='\0';
    if( firsttime ) { firsttime=0; buffpnt=sendbuff;}
5    else
    {
        /* set up shell cmd line to insert to mysql!! */
        systemstring[0]='\0';
        strcat(systemstring,"echo 'insert into raw (str2,stamped) values (\");
10        strcat(systemstring, field);

        /* loop over data and make comma delimited*/
        strcat(systemstring, " ");

15        for(i=0; i<strlen(sendbuff); i++)
        {
            sprintf(outb,"%d",(int)sendbuff[i]);
            strcat(systemstring, outb);
            strcat(systemstring, ",");
20        }

        /* echo 'insert into raw (str2,stamped) values ("2 34,23,"now()') | /usr/www/local/mysql-
        3.22.27/bin/mysql caption */

25    /* strcat(systemstring, "\",now()') >> /tmp/yyyy");    */

        strcat(systemstring, "\",now()') | /usr/www/local/mysql-3.22.27/bin/mysql
caption");
30

    /* comment out logic below to send data if also blank */
    if( strlen(sendbuff) > 0 )
        system(systemstring);

35    buffpnt=sendbuff;
} /* end else */

switch ((char)(buf[0]))
40    {
        case 0x01: field[0]='1'; break;
        case 0x02: field[0]='2'; break;
        case 0x11: field[0]='1'; break;
        case 0x12: field[0]='2'; break;
45        default: exit(0); break;
    }
```

```

    } /* end if EOCAP */
else /* not end of caption char */
{
    /* if(buf[0]==0x0a ) { buf[0]=0xff; exit(0); } */
5      *(buffpnt)=buf[0]; buffpnt++;
    LOG(buf);
}

} /* end len > 0 */

10      /* SYSTEM shell call - although inefficent right know, this still can write over 100 sub
string events to the DB
per second, far beyond what we need!!! */

15      /* system("date >> /tmp/yy"); */
    }

/* NOTE: 01 02 11 12 (hex) 01 02 start of data 11 12 no data
0x0a just means nothing ? */
20
}

```

File: cgic.c

```

5  #if CGICDEBUG
    #define CGICDEBUGSTART \
        {\
            FILE *dout; \
            dout = fopen("/home/boutell/public_html/debug", "a"); \

10  #define CGICDEBUGEND \
        fclose(dout); \
    }
    #else /* CGICDEBUG */
    #define CGICDEBUGSTART
15  #define CGICDEBUGEND
    #endif /* CGICDEBUG */

    #include <stdio.h>
    #include <string.h>
20  #include <ctype.h>
    #include <stdlib.h>
    #ifndef NO_UNISTD
    #include <unistd.h>
    #endif /* NO_UNISTD */
25  #include "/usr/local/cgic103/cgic.h"

    #define cgiStrEq(a, b) (!strcmp((a), (b)))

    char *cgiServerSoftware;
30  char *cgiServerName;
    char *cgiGatewayInterface;
    char *cgiServerProtocol;
    char *cgiServerPort;
    char *cgiRequestMethod;
35  char *cgiPathInfo;
    char *cgiPathTranslated;
    char *cgiScriptName;
    char *cgiQueryString;
    char *cgiRemoteHost;
40  char *cgiRemoteAddr;
    char *cgiAuthType;
    char *cgiRemoteUser;
    char *cgiRemoteIdent;
    char *cgiContentType;
45  int cgiContentLength;
    char *cgiAccept;
```



```
char *cgiUserAgent;

FILE *cgiIn;
FILE *cgiOut;
5  /* One form entry, consisting of an attribute-value pair. */

typedef struct cgiFormEntryStruct {
    char *attr;
10    char *value;
    struct cgiFormEntryStruct *next;
} cgiFormEntry;

/* The first form entry. */
15 static cgiFormEntry *cgiFormEntryFirst;

/* True if CGI environment was restored from a file. */
static int cgiRestored = 0;

20 static void cgiGetenv(char **s, char *var);

typedef enum {
    cgiParseSuccess,
    cgiParseMemory,
25    cgiParseIO
} cgiParseResultType;

static cgiParseResultType cgiParseGetFormInput();
static cgiParseResultType cgiParsePostFormInput();
30 static cgiParseResultType cgiParseFormInput(char *data, int length);
static void cgiSetupConstants();
static void cgiFreeResources();
static int cgiStrEqNc(char *s1, char *s2);

35 int main(int argc, char *argv[]) {
    int result;
    char *cgiContentLengthString;
    cgiSetupConstants();
    cgiGetenv(&cgiServerSoftware, "SERVER_SOFTWARE");
40    cgiGetenv(&cgiServerName, "SERVER_NAME");
    cgiGetenv(&cgiGatewayInterface, "GATEWAY_INTERFACE");
    cgiGetenv(&cgiServerProtocol, "SERVER_PROTOCOL");
    cgiGetenv(&cgiServerPort, "SERVER_PORT");
    cgiGetenv(&cgiRequestMethod, "REQUEST_METHOD");
45    cgiGetenv(&cgiPathInfo, "PATH_INFO");
    cgiGetenv(&cgiPathTranslated, "PATH_TRANSLATED");
```

```

cgiGetenv(&cgiScriptName, "SCRIPT_NAME");
cgiGetenv(&cgiQueryString, "QUERY_STRING");
cgiGetenv(&cgiRemoteHost, "REMOTE_HOST");
cgiGetenv(&cgiRemoteAddr, "REMOTE_ADDR");
5  cgiGetenv(&cgiAuthType, "AUTH_TYPE");
    cgiGetenv(&cgiRemoteUser, "REMOTE_USER");
    cgiGetenv(&cgiRemoteIdent, "REMOTE_IDENT");
    cgiGetenv(&cgiContentType, "CONTENT_TYPE");
    cgiGetenv(&cgiContentLengthString, "CONTENT_LENGTH");
10  cgiContentLength = atoi(cgiContentLengthString);
    cgiGetenv(&cgiAccept, "HTTP_ACCEPT");
    cgiGetenv(&cgiUserAgent, "HTTP_USER_AGENT");
#ifdef CGICDEBUG
    CGICDEBUGSTART
15  fprintf(dout, "%d\n", cgiContentLength);
    fprintf(dout, "%s\n", cgiRequestMethod);
    fprintf(dout, "%s\n", cgiContentType);
    CGICDEBUGEND
#endif /* CGICDEBUG */
20  cgiFormEntryFirst = 0;
    cgiIn = stdin;
    cgiOut = stdout;
    cgiRestored = 0;

25  /* These five lines keep compilers from
        producing warnings that argc and argv
        are unused. They have no actual function. */
    if (argc) {
30      if (argv[0]) {
          cgiRestored = 0;
      }
    }

35  if (cgiStrEqNc(cgiRequestMethod, "post")) {
#ifdef CGICDEBUG
    CGICDEBUGSTART
    fprintf(dout, "POST recognized\n");
40  CGICDEBUGEND
#endif /* CGICDEBUG */
    if (cgiStrEqNc(cgiContentType, "application/x-www-form-urlencoded")) {

#ifdef CGICDEBUG
45  CGICDEBUGSTART
    fprintf(dout, "Calling PostFormInput\n");
```

```

                                CGICDEBUGEND
#endif /* CGICDEBUG */
                                if (cgiParsePostFormInput() != cgiParseSuccess) {
#ifdef CGICDEBUG
5                                CGICDEBUGSTART
                                fprintf(dout, "PostFormInput failed\n");
                                CGICDEBUGEND
                                #endif /* CGICDEBUG */
                                cgiFreeResources();
10                                return -1;
                                }
                                #ifdef CGICDEBUG
                                CGICDEBUGSTART
                                fprintf(dout, "PostFormInput succeeded\n");
15                                CGICDEBUGEND
                                #endif /* CGICDEBUG */
                                }
                                } else if (cgiStrEqNc(cgiRequestMethod, "get")) {
                                /* The spec says this should be taken care of by
20                                the server, but... it isn't */
                                cgiContentLength = strlen(cgiQueryString);
                                if (cgiParseGetFormInput() != cgiParseSuccess) {
                                #ifdef CGICDEBUG
                                CGICDEBUGSTART
25                                fprintf(dout, "GetFormInput failed\n");
                                CGICDEBUGEND
                                #endif /* CGICDEBUG */
                                cgiFreeResources();
                                return -1;
30                                } else {
                                #ifdef CGICDEBUG
                                CGICDEBUGSTART
                                fprintf(dout, "GetFormInput succeeded\n");
                                CGICDEBUGEND
35                                #endif /* CGICDEBUG */
                                }
                                }
                                result = cgiMain();
                                cgiFreeResources();
40
                                /* The following two lines allegedly produce better behavior
                                when run with the CERN server on some platforms.
                                Remove them and experiment if you wish. */
                                fflush(stdout);
45                                sleep(1);
                                return result;
```

```

    }

    static void cgiGetenv(char **s, char *var){
        *s = getenv(var);
5        if (!(*s)) {
            *s = "";
        }
    }

10    static cgiParseResultType cgiParsePostFormInput() {
        char *input;
        cgiParseResultType result;
        if (!cgiContentLength) {
            return cgiParseSuccess;
15        }
        input = (char *) malloc(cgiContentLength);
        if (!input) {
            return cgiParseMemory;
        }
20        if (fread(input, 1, cgiContentLength, cgiIn) != cgiContentLength) {
            return cgiParseIO;
        }
        result = cgiParseFormInput(input, cgiContentLength);
        free(input);
25        return result;
    }

    static cgiParseResultType cgiParseGetFormInput() {
        return cgiParseFormInput(cgiQueryString, cgiContentLength);
30    }

    typedef enum {
        cgiEscapeRest,
        cgiEscapeFirst,
35        cgiEscapeSecond
    } cgiEscapeState;

    typedef enum {
        cgiUnescapeSuccess,
40        cgiUnescapeMemory
    } cgiUnescapeResultType;

    static cgiUnescapeResultType cgiUnescapeChars(char **sp, char *cp, int len);

45    static cgiParseResultType cgiParseFormInput(char *data, int length) {
        /* Scan for pairs, unescaping and storing them as they are found. */

```

```
int pos = 0;
cgiFormEntry *n;
cgiFormEntry *l = 0;
while (pos != length) {
5       int foundEq = 0;
        int foundAmp = 0;
        int start = pos;
        int len = 0;
        char *attr;
10       char *value;
        while (pos != length) {
            if (data[pos] == '=') {
                foundEq = 1;
                pos++;
                break;
15         }
            pos++;
            len++;
        }
        if (!foundEq) {
20         break;
        }
        if (cgiUnescapeChars(&attr, data+start, len)
            != cgiUnescapeSuccess) {
25         return cgiParseMemory;
        }
        start = pos;
        len = 0;
        while (pos != length) {
30         if (data[pos] == '&') {
            foundAmp = 1;
            pos++;
            break;
        }
        pos++;
35         len++;
    }
    /* The last pair probably won't be followed by a &, but
       that's fine, so check for that after accepting it */
40     if (cgiUnescapeChars(&value, data+start, len)
        != cgiUnescapeSuccess) {
        return cgiParseMemory;
    }
    /* OK, we have a new pair, add it to the list. */
45     n = (cgiFormEntry *) malloc(sizeof(cgiFormEntry));
    if (!n) {
```

```
        return cgiParseMemory;
    }
    n->attr = attr;
    n->value = value;
5    n->next = 0;
    if (!l) {
        cgiFormEntryFirst = n;
    } else {
        l->next = n;
10    }
    l = n;
    if (!foundAmp) {
        break;
    }
15    }
    return cgiParseSuccess;
}

static int cgiHexValue[256];
20
cgiUnescapeResultType cgiUnescapeChars(char **sp, char *cp, int len) {
    char *s;
    cgiEscapeState escapeState = cgiEscapeRest;
    int escapedValue = 0;
25    int srcPos = 0;
    int dstPos = 0;
    s = (char *) malloc(len + 1);
    if (!s) {
        return cgiUnescapeMemory;
30    }
    while (srcPos < len) {
        int ch = cp[srcPos];
        switch (escapeState) {
            case cgiEscapeRest:
35                if (ch == '%') {
                    escapeState = cgiEscapeFirst;
                } else if (ch == '+') {
                    s[dstPos++] = ' ';
                } else {
40                    s[dstPos++] = ch;
                }
                break;
            case cgiEscapeFirst:
                escapedValue = cgiHexValue[ch] << 4;
45                escapeState = cgiEscapeSecond;
                break;
```

```

        case cgiEscapeSecond:
            escapedValue += cgiHexValue[ch];
            s[dstPos++] = escapedValue;
            escapeState = cgiEscapeRest;
5             break;
        }
        srcPos++;
    }
    s[dstPos] = '\0';
10    *sp = s;
    return cgiUnescapeSuccess;
}

static void cgiSetupConstants() {
15    int i;
    for (i=0; (i < 256); i++) {
        cgiHexValue[i] = 0;
    }
    cgiHexValue['0'] = 0;
20    cgiHexValue['1'] = 1;
    cgiHexValue['2'] = 2;
    cgiHexValue['3'] = 3;
    cgiHexValue['4'] = 4;
    cgiHexValue['5'] = 5;
25    cgiHexValue['6'] = 6;
    cgiHexValue['7'] = 7;
    cgiHexValue['8'] = 8;
    cgiHexValue['9'] = 9;
    cgiHexValue['A'] = 10;
30    cgiHexValue['B'] = 11;
    cgiHexValue['C'] = 12;
    cgiHexValue['D'] = 13;
    cgiHexValue['E'] = 14;
    cgiHexValue['F'] = 15;
35    cgiHexValue['a'] = 10;
    cgiHexValue['b'] = 11;
    cgiHexValue['c'] = 12;
    cgiHexValue['d'] = 13;
    cgiHexValue['e'] = 14;
40    cgiHexValue['f'] = 15;
}

static void cgiFreeResources() {
    cgiFormEntry *c = cgiFormEntryFirst;
45    cgiFormEntry *n;
    while (c) {
```

```

        n = c->next;
        free(c->attr);
        free(c->value);
        free(c);
5      c = n;
    }
    /* If the cgi environment was restored from a saved environment,
       then these are in allocated space and must also be freed */
    if (cgiRestored) {
10      free(cgiServerSoftware);
        free(cgiServerName);
        free(cgiGatewayInterface);
        free(cgiServerProtocol);
        free(cgiServerPort);
15      free(cgiRequestMethod);
        free(cgiPathInfo);
        free(cgiPathTranslated);
        free(cgiScriptName);
        free(cgiQueryString);
20      free(cgiRemoteHost);
        free(cgiRemoteAddr);
        free(cgiAuthType);
        free(cgiRemoteUser);
        free(cgiRemoteIdent);
25      free(cgiContentType);
        free(cgiAccept);
        free(cgiUserAgent);
    }
    }
30
    static cgiFormResultType cgiFormEntryString(
        cgiFormEntry *e, char *result, int max, int newlines);

    static cgiFormEntry *cgiFormEntryFindFirst(char *name);
35    static cgiFormEntry *cgiFormEntryFindNext();

    cgiFormResultType cgiFormString(
        char *name, char *result, int max) {
        cgiFormEntry *e;
40      e = cgiFormEntryFindFirst(name);
        if (!e) {
            strcpy(result, "");
            return cgiFormNotFound;
        }
45      return cgiFormEntryString(e, result, max, 1);
    }
}
```



```
cgiFormResultType cgiFormStringNoNewlines(
    char *name, char *result, int max) {
    cgiFormEntry *e;
5    e = cgiFormEntryFindFirst(name);
    if (!e) {
        strcpy(result, "");
        return cgiFormNotFound;
    }
10    return cgiFormEntryString(e, result, max, 0);
}

cgiFormResultType cgiFormStringMultiple(
    char *name, char ***result) {
15    char **stringArray;
    cgiFormEntry *e;
    int i;
    int total = 0;
    /* Make two passes. One would be more efficient, but this
20    function is not commonly used. The select menu and
    radio box functions are faster. */
    e = cgiFormEntryFindFirst(name);
    if (e != 0) {
        do {
25            total++;
        } while ((e = cgiFormEntryFindNext()) != 0);
    }
    stringArray = (char **) malloc(sizeof(char *) * (total + 1));
    if (!stringArray) {
30        *result = 0;
        return cgiFormMemory;
    }
    /* initialize all entries to null; the last will stay that way */
    for (i=0; (i <= total); i++) {
35        stringArray[i] = 0;
    }
    /* Now go get the entries */
    e = cgiFormEntryFindFirst(name);
#ifdef CGICDEBUG
40    CGICDEBUGSTART
    fprintf(dout, "StringMultiple Beginning\n");
    CGICDEBUGEND
#endif /* CGICDEBUG */
    if (e) {
45        i = 0;
        do {
```

```
int max = strlen(e->value) + 1;
stringArray[i] = (char *) malloc(max);
if (stringArray[i] == 0) {
    /* Memory problems */
5    cgiStringArrayFree(stringArray);
    *result = 0;
    return cgiFormMemory;
}
strcpy(stringArray[i], e->value);
10    cgiFormEntryString(e, stringArray[i], max, 1);
    i++;
} while ((e = cgiFormEntryFindNext()) != 0);
*result = stringArray;
#ifdef CGICDEBUG
15    CGICDEBUGSTART
    fprintf(dout, "StringMultiple Succeeding\n");
    CGICDEBUGEND
#endif /* CGICDEBUG */
    return cgiFormSuccess;
20    } else {
        *result = stringArray;
#ifdef CGICDEBUG
        CGICDEBUGSTART
        fprintf(dout, "StringMultiple found nothing\n");
        CGICDEBUGEND
25    #endif /* CGICDEBUG */
        return cgiFormNotFound;
    }
}
30
cgiFormResultType cgiFormStringSpaceNeeded(
    char *name, int *result) {
    cgiFormEntry *e;
    e = cgiFormEntryFindFirst(name);
35    if (!e) {
        *result = 1;
        return cgiFormNotFound;
    }
    *result = strlen(e->value) + 1;
40    return cgiFormSuccess;
}

static cgiFormResultType cgiFormEntryString(
45    cgiFormEntry *e, char *result, int max, int newlines) {
    char *dp, *sp;
    int truncated = 0;
```

```
int len = 0;
int avail = max-1;
int crCount = 0;
int lfCount = 0;
5 dp = result;
  sp = e->value;
  while (1) {
    int ch;
    ch = *sp;
10    if (len == avail) {
      truncated = 1;
      break;
    }
    /* Fix the CR/LF, LF, CR nightmare: watch for
15    consecutive bursts of CRs and LFs in whatever
    pattern, then actually output the larger number
    of LFs. Consistently sane, yet it still allows
    consecutive blank lines when the user
    actually intends them. */
20    if ((ch == 13) || (ch == 10)) {
      if (ch == 13) {
        crCount++;
      } else {
        lfCount++;
      }
25    } else {
      if (crCount || lfCount) {
        int lfsAdd = crCount;
        if (lfCount > crCount) {
30          lfsAdd = lfCount;
        }
        /* Stomp all newlines if desired */
        if (!newlines) {
          lfsAdd = 0;
35        }
        while (lfsAdd) {
          if (len == avail) {
            truncated = 1;
            break;
40          }
          *dp = 10;
          dp++;
          lfsAdd--;
          len++;
45        }
        crCount = 0;
      }
    }
  }
```

```
        ifCount = 0;
    }
    if (ch == '\0') {
        /* The end of the source string */
5        break;
    }
    *dp = ch;
    dp++;
    len++;
10    }
    sp++;
}
*dp = '\0';
if (truncated) {
15    return cgiFormTruncated;
} else if (!len) {
    return cgiFormEmpty;
} else {
    return cgiFormSuccess;
20 }
}
```

```
static int cgiFirstNonspaceChar(char *s);
```

```
25 cgiFormResultType cgiFormInteger(
    char *name, int *result, int defaultV) {
    cgiFormEntry *e;
    int ch;
    e = cgiFormEntryFindFirst(name);
30    if (!e) {
        *result = defaultV;
        return cgiFormNotFound;
    }
    if (!strlen(e->value)) {
35        *result = defaultV;
        return cgiFormEmpty;
    }
    ch = cgiFirstNonspaceChar(e->value);
    if (!(isdigit(ch)) && (ch != '-') && (ch != '+')) {
40        *result = defaultV;
        return cgiFormBadType;
    } else {
        *result = atoi(e->value);
        return cgiFormSuccess;
45    }
}
```

```
cgiFormResultType cgiFormIntegerBounded(
    char *name, int *result, int min, int max, int defaultV) {
    cgiFormResultType error = cgiFormInteger(name, result, defaultV);
5    if (error != cgiFormSuccess) {
        return error;
    }
    if (*result < min) {
        *result = min;
10    return cgiFormConstrained;
    }
    if (*result > max) {
        *result = max;
        return cgiFormConstrained;
15    }
    return cgiFormSuccess;
}

cgiFormResultType cgiFormDouble(
20    char *name, double *result, double defaultV) {
    cgiFormEntry *e;
    int ch;
    e = cgiFormEntryFindFirst(name);
    if (!e) {
25        *result = defaultV;
        return cgiFormNotFound;
    }
    if (!strlen(e->value)) {
        *result = defaultV;
30    return cgiFormEmpty;
    }
    ch = cgiFirstNonspaceChar(e->value);
    if (!(isdigit(ch) && (ch != '.') && (ch != '-') && (ch != '+'))) {
        *result = defaultV;
35    return cgiFormBadType;
    } else {
        *result = atof(e->value);
        return cgiFormSuccess;
    }
40 }

cgiFormResultType cgiFormDoubleBounded(
    char *name, double *result, double min, double max, double defaultV) {
    cgiFormResultType error = cgiFormDouble(name, result, defaultV);
45    if (error != cgiFormSuccess) {
        return error;
```

```

    }
    if (*result < min) {
        *result = min;
        return cgiFormConstrained;
5    }
    if (*result > max) {
        *result = max;
        return cgiFormConstrained;
    }
10    return cgiFormSuccess;
}

cgiFormResultType cgiFormSelectSingle(
    char *name, char **choicesText, int choicesTotal,
15    int *result, int defaultV)
{
    cgiFormEntry *e;
    int i;
    e = cgiFormEntryFindFirst(name);
20    #ifdef CGICDEBUG
        CGICDEBUGSTART
        fprintf(dout, "%d\n", (int) e);
        CGICDEBUGEND
    #endif /* CGICDEBUG */
    if (!e) {
25        *result = defaultV;
        return cgiFormNotFound;
    }
    for (i=0; (i < choicesTotal); i++) {
30    #ifdef CGICDEBUG
        CGICDEBUGSTART
        fprintf(dout, "%s %s\n", choicesText[i], e->value);
        CGICDEBUGEND
    #endif /* CGICDEBUG */
    if (cgiStrEq(choicesText[i], e->value)) {
35    #ifdef CGICDEBUG
        CGICDEBUGSTART
        fprintf(dout, "MATCH\n");
        CGICDEBUGEND
    #endif /* CGICDEBUG */
        *result = i;
        return cgiFormSuccess;
    }
}
45    *result = defaultV;
    return cgiFormNoSuchChoice;
}
```

```

    }

cgiFormResultType cgiFormSelectMultiple(
    char *name, char **choicesText, int choicesTotal,
5    int *result, int *invalid)
{
    cgiFormEntry *e;
    int i;
    int hits = 0;
    int invalidE = 0;
10    for (i=0; (i < choicesTotal); i++) {
        result[i] = 0;
    }
    e = cgiFormEntryFindFirst(name);
15    if (!e) {
        *invalid = invalidE;
        return cgiFormNotFound;
    }
    do {
20        int hit = 0;
        for (i=0; (i < choicesTotal); i++) {
            if (cgiStrEq(choicesText[i], e->value)) {
                result[i] = 1;
                hits++;
                hit = 1;
                break;
            }
        }
        if (!(hit)) {
30            invalidE++;
        }
    } while ((e = cgiFormEntryFindNext()) != 0);

    *invalid = invalidE;
35    if (hits) {
        return cgiFormSuccess;
    } else {
        return cgiFormNotFound;
40    }
}

cgiFormResultType cgiFormCheckboxSingle(
    char *name)
45 {
    cgiFormEntry *e;
```

```

    e = cgiFormEntryFindFirst(name);
    if (!e) {
        return cgiFormNotFound;
    }
5   return cgiFormSuccess;
}

extern cgiFormResultType cgiFormCheckboxMultiple(
10   char *name, char **valuesText, int valuesTotal,
    int *result, int *invalid)
{
    /* Implementation is identical to cgiFormSelectMultiple. */
    return cgiFormSelectMultiple(name, valuesText,
        valuesTotal, result, invalid);
15 }

cgiFormResultType cgiFormRadio(
    char *name,
    char **valuesText, int valuesTotal, int *result, int defaultV)
20 {
    /* Implementation is identical to cgiFormSelectSingle. */
    return cgiFormSelectSingle(name, valuesText, valuesTotal,
        result, defaultV);
}
25

void cgiHeaderLocation(char *redirectUrl) {
    fprintf(cgiOut, "Location: %s%c%c", redirectUrl, 10, 10);
}

30 void cgiHeaderStatus(int status, char *statusMessage) {
    fprintf(cgiOut, "Status: %d %s%c%c", status, statusMessage,
        10, 10);
}

35 void cgiHeaderContentType(char *mimeType) {
    fprintf(cgiOut, "Content-type: %s%c%c", mimeType, 10, 10);
}

#ifdef NO_SYSTEM
40
int cgiSaferSystem(char *command) {
    char *s;
    char *sp;
    int i;
45   int len = (strlen(command) * 2) + 1;
    s = (char *) malloc(len);
```



```

    if (!s) {
        return -1;
    }
    sp = s;
5   for (i=0; (i < len); i++) {
        if (command[i] == ';') {
            *sp = '\\';
            sp++;
        } else if (command[i] == '|') {
10          *sp = '\\';
            sp++;
        }
        *sp = command[i];
        sp++;
15    }
    *sp = '\\0';
    return system(s);
}

20 #endif /* NO_SYSTEM */

static int cgiWriteString(FILE *out, char *s);

static int cgiWriteInt(FILE *out, int i);
25 cgiEnvironmentResultType cgiWriteEnvironment(char *filename) {
    FILE *out;
    cgiFormEntry *e;
    /* Be sure to open in binary mode */
30    out = fopen(filename, "wb");
    if (!out) {
        /* Can't create file */
        return cgiEnvironmentIO;
    }
35    if (!cgiWriteString(out, cgiServerSoftware)) {
        goto error;
    }
    if (!cgiWriteString(out, cgiServerName)) {
        goto error;
    }
40    if (!cgiWriteString(out, cgiGatewayInterface)) {
        goto error;
    }
    if (!cgiWriteString(out, cgiServerProtocol)) {
45        goto error;
    }
}
```

```
if (!cgiWriteString(out, cgiServerPort)) {
    goto error;
}
if (!cgiWriteString(out, cgiRequestMethod)) {
5    goto error;
}
if (!cgiWriteString(out, cgiPathInfo)) {
    goto error;
}
10 if (!cgiWriteString(out, cgiPathTranslated)) {
    goto error;
}
if (!cgiWriteString(out, cgiScriptName)) {
    goto error;
15 }
if (!cgiWriteString(out, cgiQueryString)) {
    goto error;
}
if (!cgiWriteString(out, cgiRemoteHost)) {
20    goto error;
}
if (!cgiWriteString(out, cgiRemoteAddr)) {
    goto error;
}
25 if (!cgiWriteString(out, cgiAuthType)) {
    goto error;
}
if (!cgiWriteString(out, cgiRemoteUser)) {
    goto error;
30 }
if (!cgiWriteString(out, cgiRemoteIdent)) {
    goto error;
}
if (!cgiWriteString(out, cgiContentType)) {
35    goto error;
}
if (!cgiWriteString(out, cgiAccept)) {
    goto error;
}
40 if (!cgiWriteString(out, cgiUserAgent)) {
    goto error;
}
if (!cgiWriteInt(out, cgiContentLength)) {
    goto error;
45 }
e = cgiFormEntryFirst;
```

```
while (e) {
    if (!cgiWriteString(out, e->attr)) {
        goto error;
    }
5    if (!cgiWriteString(out, e->value)) {
        goto error;
    }
    e = e->next;
}
10 fclose(out);
return cgiEnvironmentSuccess;

error:
    fclose(out);
    /* If this function is not defined in your system,
15     you must substitute the appropriate
       file-deletion function. */
    unlink(filename);
    return cgiEnvironmentIO;
}

20 static int cgiWriteString(FILE *out, char *s) {
    int len = strlen(s);
    cgiWriteInt(out, len);
    if (fwrite(s, 1, len, out) != len) {
25         return 0;
    }
    return 1;
}

30 static int cgiWriteInt(FILE *out, int i) {
    if (!fwrite(&i, sizeof(int), 1, out)) {
        return 0;
    }
    return 1;
35 }

static int cgiReadString(FILE *out, char **s);

static int cgiReadInt(FILE *out, int *i);

40 cgiEnvironmentResultType cgiReadEnvironment(char *filename) {
    FILE *in;
    cgiFormEntry *e, *p;
    /* Free any existing data first */
45    cgiFreeResources();
    /* Be sure to open in binary mode */
```

```
in = fopen(filename, "rb");
if (!in) {
    /* Can't access file */
    return cgiEnvironmentIO;
5   }
    if (!cgiReadString(in, &cgiServerSoftware)) {
        goto error;
    }
    if (!cgiReadString(in, &cgiServerName)) {
10   goto error;
    }
    if (!cgiReadString(in, &cgiGatewayInterface)) {
        goto error;
    }
15   if (!cgiReadString(in, &cgiServerProtocol)) {
        goto error;
    }
    if (!cgiReadString(in, &cgiServerPort)) {
        goto error;
20   }
    if (!cgiReadString(in, &cgiRequestMethod)) {
        goto error;
    }
    if (!cgiReadString(in, &cgiPathInfo)) {
25   goto error;
    }
    if (!cgiReadString(in, &cgiPathTranslated)) {
        goto error;
    }
30   if (!cgiReadString(in, &cgiScriptName)) {
        goto error;
    }
    if (!cgiReadString(in, &cgiQueryString)) {
        goto error;
35   }
    if (!cgiReadString(in, &cgiRemoteHost)) {
        goto error;
    }
    if (!cgiReadString(in, &cgiRemoteAddr)) {
40   goto error;
    }
    if (!cgiReadString(in, &cgiAuthType)) {
        goto error;
    }
45   if (!cgiReadString(in, &cgiRemoteUser)) {
        goto error;
```

```

    }
    if (!cgiReadString(in, &cgiRemoteIdent)) {
        goto error;
    }
5   if (!cgiReadString(in, &cgiContentType)) {
        goto error;
    }
    if (!cgiReadString(in, &cgiAccept)) {
        goto error;
10  }
    if (!cgiReadString(in, &cgiUserAgent)) {
        goto error;
    }
    if (!cgiReadInt(in, &cgiContentLength)) {
15  goto error;
    }
    p = 0;
    while (1) {
        e = (cgiFormEntry *) malloc(sizeof(cgiFormEntry));
20     if (!e) {
            cgiFreeResources();
            fclose(in);
            return cgiEnvironmentMemory;
        }
25     if (!cgiReadString(in, &e->attr)) {
        /* This means we've reached the end of the list. */
        free(e);
        break;
    }
30     if (!cgiReadString(in, &e->value)) {
        free(e);
        goto error;
    }
    e->next = 0;
35     if (p) {
        p->next = e;
    } else {
        cgiFormEntryFirst = e;
    }
40     p = e;
    }
    fclose(in);
    cgiRestored = 1;
    return cgiEnvironmentSuccess;
45 error:
    cgiFreeResources();

```

```
        fclose(in);
        return cgiEnvironmentIO;
    }

5   static int cgiReadString(FILE *in, char **s) {
        int len;
        cgiReadInt(in, &len);
        *s = (char *) malloc(len + 1);
        if (!(*s)) {
10            return 0;
        }
        if (fread(*s, 1, len, in) != len) {
            return 0;
        }
15        (*s)[len] = '\0';
        return 1;
    }

    static int cgiReadInt(FILE *out, int *i) {
20        if (!fread(i, sizeof(int), 1, out)) {
            return 0;
        }
        return 1;
    }

25   static int cgiStrEqNc(char *s1, char *s2) {
        while(1) {
            if (!(*s1)) {
                if (!(*s2)) {
30                    return 1;
                } else {
                    return 0;
                }
            } else if (!(*s2)) {
35                return 0;
            }
            if (isalpha(*s1)) {
                if (tolower(*s1) != tolower(*s2)) {
40                    return 0;
                }
            } else if ((*s1) != (*s2)) {
                return 0;
            }
            s1++;
45            s2++;
        }
    }
```

```

    }

    static char *cgiFindTarget = 0;
    static cgiFormEntry *cgiFindPos = 0;
5
    static cgiFormEntry *cgiFormEntryFindFirst(char *name) {
        cgiFindTarget = name;
        cgiFindPos = cgiFormEntryFirst;
        return cgiFormEntryFindNext();
10    }

    static cgiFormEntry *cgiFormEntryFindNext() {
        while (cgiFindPos) {
            cgiFormEntry *c = cgiFindPos;
15            cgiFindPos = c->next;
            if (!strcmp(c->attr, cgiFindTarget)) {
                return c;
            }
        }
20        return 0;
    }

    static int cgiFirstNonspaceChar(char *s) {
        int len = strspn(s, " \n\r\t");
25        return s[len];
    }

    void cgiStringArrayFree(char **stringArray) {
        char *p;
30        p = *stringArray;
        while (p) {
            free(p);
            stringArray++;
            p = *stringArray;
35        }
    }
}
```

File: listen.c

```
5  #include <stdio.h>
   #include <sys/types.h>
   #include <sys/socket.h>
   #include <linux/in.h>
   #include <netdb.h>

10  /*
   Set captionsocket in BSD services file to a socketnumber
   9999 for example
   */

15  /* Relay across socket to Manfred's access,
   could be simple socket, httpd on socket 80,
   or even fake an odbc!!! since we need only select!!! */

20  struct sockaddr_in theaddr = { AF_INET };

   socket_listen()
   {
25      struct servent *serv, *getservbyname();
      struct hostent *host, *gethostbyname();
      int ss, ns, length;
      char ipbuf[44];

30      if( (ss=socket(AF_INET, SOCK_STREAM, 0)) < 0 )
      {
          puts("\nsocket() failed.");
          return(0);
      }

35      /* look up the address of the local host */
      host = gethostbyname("localhost");
      if( host == (struct hostent *)0 )
      {
40          puts("\nFailed to get host ip address for this name: \"caption\"");
          return(0);
      }
      bcopy(host->h_addr, (char *)&theaddr, host->h_length);

45      theaddr.sin_family = AF_INET;
```



```
length = sizeof(theaddr);
if (getsockname(ss, (struct sockaddr *)&theaddr, &length) == -1)
{
    close(ss);
5    puts("Bad address for socket.");
    return(0);
}

10    if( (serv = getservbyname("captionsocket","tcp")) == 0 )
    {
        close(ss);
        puts("/etc/services does not have socket caption registered.");
        return(0);
    }
15    /*theaddr.sin_port = 0;*/
    theaddr.sin_port = serv->s_port;

    if( bind(ss, (struct sockaddr *)&theaddr, sizeof(theaddr)) < 0)
    {
20        puts("\nbind() failed.");
        close(ss);
        return(0);
    }

25    if( listen(ss,(int)1) < 0 )
    {
        puts("\nlisten() failed.");
        return(0);
    }

30    if( (ns=accept(ss, 0, 0)) < 0 )
    {
        puts("\naccept() failed.");
        return(0);
    }
35    close(ss);

    return(ns);
40 }
```

File: Makefile

```
5  #You may wish to add -DNO_UNISTD and -DNO_SYSTEM to CFLAGS if
   #your system lacks the /usr/include/unistd.h header file
   #or the system() function, respectively.

   CFLAGS      = -O -DHUU -DAUTOCONNECT
   LDFLAGS     = -s

10

   $(NATEXES):$$@.c
       $(CC) $(CFLAGS) -o $$@ $? $(LDFLAGS)

15  all: captionsock

   captionsock: captionsock.o
       gcc captionsock.o -o captionsock
```